



**IMPLEMENTACIÓN DE PROCESOS DE NEGOCIO A  
TRAVÉS DE SERVICIOS APLICANDO METAMODELOS,  
SOFTWARE DISTRIBUIDO Y ASPECTOS SOCIALES**

TESISTA

**PATRICIA BAZÁN**

TESIS PRESENTADA PARA OBTENER EL GRADO DE DOCTOR EN CIENCIAS  
INFORMÁTICAS

**FACULTAD DE INFORMÁTICA  
UNIVERSIDAD NACIONAL DE LA PLATA**

FEBRERO, 2015



## **PREFACIO**

Esta Tesis es presentada como parte de los requisitos para optar al grado académico de Doctor en Ciencias Informáticas de la Facultad de Informática de la Universidad Nacional de La Plata, La Plata, Buenos Aires, Argentina y no ha sido presentada previamente para la obtención de otro título en esta Universidad u otras. La misma contiene los resultados obtenidos en investigaciones llevadas a cabo en el Laboratorio de Investigación en Nuevas Tecnologías Informáticas (LINTI) de la Facultad de Informática, de la Universidad Nacional de La Plata.

El trabajo se realizó entre mayo de 2011 y diciembre de 2014, bajo la dirección de la Dra. Roxana Giandini y la Dra. Elsa Estevez.

Mg. Patricia Bazán

La Plata, Febrero de 2015

LINTI – Laboratorio de Investigación en Nuevas Tecnologías Informáticas  
Facultad de Informática  
Universidad Nacional de La Plata

## AGRADECIMIENTOS

Este trabajo ha sido la concreción de un sueño personal que se inició allá por 1984, cuando ingresé a la carrera de grado y no había posibilidades de doctorarse en la Argentina. Mi anhelo era llegar al escalón más alto al que la vida académica me desafiaba. En el trayecto muchas cosas sucedieron, entre ellas la creación de carreras de postgrado vinculadas a la informática en la Universidad Nacional de La Plata y luego la creación de la Facultad de Informática, primera y única en el país. Con este escenario, era un deber conmigo misma poder concretar este sueño, estaba ahí, al alcance de mi mano, en mi mismo ámbito laboral y contando con un excelente nivel académico, solamente había que poner manos a la obra.

Quiero agradecer infinitamente a mis directoras - Roxana Giandini y Elsa Estevez, por su confianza, su apoyo y su acompañamiento en todo momento. Al director del LINTI, Javier Diaz, a quien conozco desde que me inicié como ayudante alumna, porque siempre ha depositado en mí una gran confianza y me ha alentado a llevar a cabo este proyecto. Finalmente, quiero agradecer a las dos personas que más amo en este mundo, mis hijas, Sabrina y Candela, que me acompañan y me sostienen, que me alientan y me hacen feliz. Sin ellas, nada de esto vale la pena. No me voy a olvidar de mi ángel guardián, Santiago, que siempre posado en mi hombro ha susurrado más de un párrafo que yo no lograba desenmarañar. Y por supuesto a Dios, mi Señor, que jamás me ha abandonado.



## RESUMEN

El enfoque orientado a procesos de negocio es un aspecto ampliamente relevante para las organizaciones, que en los últimos años ha recibido importante atención de la comunidad científica. Asimismo, los avances en cuanto a la provisión de herramientas de soporte para automatizar la gestión los procesos de negocio también ha adquirido gran relevancia. Sin embargo, la brecha existente entre el área de negocio y el área de tecnología, representadas por analistas de negocios y expertos en informática, respectivamente, sigue constituyendo un escollo a la hora de aplicar una metodología de gestión por procesos de negocio dentro de las organizaciones. Por otra parte, los pocos avances tecnológicos respecto a la incorporación de nuevos modelos computacionales distribuidos y de aspectos sociales en la ejecución de los procesos de negocio y de las herramientas que la soportan, contribuyen a aumentar dicha brecha. Por estos motivos, resultan relevantes las investigaciones en metodologías, marcos de trabajo y herramientas que incluyan estos nuevos paradigmas en la administración de procesos de negocio.

En esta tesis se plantea mejorar y actualizar la Metodología Integradora de Servicios y Procesos (MISP), propuesta por la autora en su tesis de Maestría en Redes de Datos, proveyendo una nueva visión de los procesos y los servicios a la luz de los avances tecnológicos y buscando reducir la brecha entre el negocio y la tecnología. Específicamente, el trabajo se enfoca en dos principales problemas: 1) la mejora al modelado de procesos y servicios mediante la definición e integración de metamodelos, aplicados a las fases de diseño de procesos dentro del ciclo de vida de los procesos de negocio, y 2) la revisión de aspectos tecnológicos modernos – como por ejemplo, la distribución de las actividades de los procesos y la inclusión de aspectos sociales relacionados con la ejecución de los mismos, cuya aplicación resulta de interés en las etapas de despliegue, ejecución y monitoreo de los procesos de negocio.

Atendiendo los problemas descritos, esta tesis tiene dos contribuciones importantes. Por un lado, la mejora a la interacción entre procesos y servicios mediante la provisión de un lenguaje para describir servicios que integra las actividades del proceso con las componentes de software que lo implementan. Por otro lado, una especificación de requisitos y la implementación de un prototipo de herramientas que permiten incorporar aspectos de software distribuido que enriquecen los rastros de ejecución de los procesos, y características sociales a la gestión de procesos. Esta última contribución permite optimizar la fase de monitoreo del ciclo de vida de los procesos y acelerar la mejora continua de los mismos.



# ÍNDICE GENERAL

<b>INTRODUCCIÓN .....</b>	<b>1</b>
1.1. PROCESOS DE NEGOCIO.....	2
1.1.1. Definición de Conceptos .....	2
1.1.2. Orquestación y Coreografía.....	4
1.1.3. Nuevas Tendencias .....	6
1.2. PROCESOS DE NEGOCIO SOCIALES Y COLABORATIVOS .....	7
1.2.1. Debilidades de BPM Respecto de los Conceptos de Software Social .....	7
1.2.2. Software Social y BPM Social .....	8
1.3. ENUNCIADO DEL PROBLEMA .....	10
1.3.1. Mejoras Vinculadas al Modelado de Procesos y Servicios.....	10
1.3.2. Mejoras Vinculadas a la Ejecución y Despliegue.....	11
1.4. ENFOQUE DE SOLUCIÓN.....	11
1.5. CONTRIBUCIÓN DE LA TESIS .....	12
1.6. ESTRUCTURA DE LA TESIS.....	12
<b>CONCEPTOS RELACIONADOS .....</b>	<b>14</b>
2.1. GESTIÓN DE PROCESOS DE NEGOCIOS .....	14
2.1.1. Definiciones .....	14
2.1.2. Ciclo de Vida de los Procesos de Negocio .....	15
2.1.3. Modelado de Procesos - Fundamentos .....	17
2.1.4. Las Tres Dimensiones de BPM .....	18
2.1.5. Nuevos Desafíos para BPM.....	19
2.2. DESARROLLO DIRIGIDO POR MODELOS.....	20
2.2.1. Orígenes de MDD y su Aplicación a BPM .....	20
2.2.2. Arquitectura Dirigida por Modelos .....	21
2.2.3. Tipos de Modelos en MDD.....	22
2.2.4. La Arquitectura de Cuatro Capas .....	23
2.2.5. Uso del Metamodelado en MDD .....	24
2.3. COMPUTACIÓN EN LA NUBE.....	24
2.3.1. Definiciones .....	24
2.3.2. Antecedentes Tecnológicos .....	27
2.3.4. El Modelo de Nube.....	28
2.3.5. Analisis del Modelo de Nube y su Aplicación en BPM .....	30
2.4. SOFTWARE SOCIAL .....	32
2.4.1. Las Organizaciones y el Software Social .....	33
2.4.2. Web 2.0 y el Software Social .....	34
2.4.3. Evolución del Software Social.....	35
2.5. BPM Y COMPUTACIÓN EN LA NUBE .....	36
2.5.1. BPM en el Nube - Implicancias y Consideraciones .....	36
2.5.2. Condiciones para Utilizar BPM en la Nube .....	38
2.5.3. Combinación Nube y Embebido .....	39
2.5.4. Descomposición de Procesos.....	41
2.6. BPM Y SOFTWARE SOCIAL.....	42
2.6.1. El Software Social en la Fase de Diseño de Procesos de Negocio.....	43
2.6.2. El Software Social en Despliegue de Procesos de Negocio .....	43
2.6.3. Identidad Digital y Reputación en los Procesos de Negocio.....	44
2.6.4. Los Procesos de Negocio y el Procesamiento de Información .....	44
2.7. CONCLUSIONES DEL CAPITULO .....	45
<b>TRABAJOS RELACIONADOS .....</b>	<b>46</b>
3.1. LENGUAJES DE MODELADO Y WORKFLOWS .....	46
3.1.1. BPM y MDD.....	47
3.1.2. Metamodelos para Ejecutar Procesos .....	48
3.1.3. Metamodelos y Reglas de Transformación.....	48
3.1.4. MDA como Método de Desarrollo de Software .....	49
3.2. TECNOLOGÍAS.....	50
3.2.1. Servicios y su Composición.....	50
3.2.2. Disposición de Procesos y Servicios en la Nube .....	52
3.3. HERRAMIENTAS .....	54
3.4. ESTADO DEL ARTE EN SOFTWARE SOCIAL APLICADO A BPM.....	56
3.5. CONCLUSIONES.....	57



<b>METODOLOGÍA INTEGRADORA DE PROCESOS Y SERVICIOS .....</b>	<b>59</b>
4.1. INTRODUCCIÓN .....	59
4.2. MISP. DESCRIPCIÓN FOCALIZADA EN LAS ETAPAS REVISADAS .....	60
4.2.1. Descripción Detallada de la Etapa 4 - Modelado de Procesos .....	63
4.2.2. Descripción Detallada de la Etapa 5 – Modelado de Servicios.....	63
4.3. INTEGRACIÓN DE PROCESOS Y SERVICIOS - PROPUESTAS EXISTENTES.....	65
4.3.1. Servicios.....	65
4.3.2. Procesos.....	65
4.3.3. Integración entre Procesos y Servicios.....	66
4.3.4. Discusión.....	66
4.4. MISP Y LAS MEJORAS INTRODUCIDAS .....	67
4.4.1. Introducción.....	67
4.4.2. Pasos de interacción.....	70
4.4.3. Metamodelo de Conceptualización de Servicios.....	71
4.4.4. Definición de componentes de MISP .....	72
4.4.5. Integración de Metamodelos de Servicios (P2S) y Componentes (SCA) .....	<b>iError! Marcador no definido.</b>
4.4.6. De los Componentes a los Servicios Web .....	76
4.5. MISP Y SU MEJORA A TRAVÉS DE METAMODELOS. UN EJEMPLO .....	79
4.5.1. Descripción del Problema .....	79
4.5.2. Modelado del Proceso en BPMN.....	79
4.5.3. Modelado y Conceptualización de Servicios .....	81
4.5.4. Definición de Componentes.....	83
4.5.5. Discusión.....	85
4.6. CONCLUSIONES DEL CAPÍTULO .....	86
<b>BPMS SOCIALES, COLABORATIVOS Y DISTRIBUIDOS .....</b>	<b>87</b>
5.1. MOTIVACIÓN Y VISIÓN GENERAL .....	87
5.1.1. BPaaS - Business Process as a Service .....	87
5.1.2. BPM en la Nube .....	90
5.1.3. BPM en la Nube: características del modelo.....	91
5.2. PROCESOS DISTRIBUIDOS EN BPMS: SOLUCIÓN PROPUESTA.....	92
5.2.1. Motivación .....	92
5.2.2. La Ejecución Distribuida de Procesos de Negocio.....	93
5.2.3. Monitoreo de Procesos en la Nube.....	94
5.3. BPMS SOCIALES Y COLABORATIVOS: SOLUCIÓN PROPUESTA .....	100
5.3.1. Motivación .....	100
5.3.2. Especificación de Requisitos para SocialFlow .....	101
5.3.3. Un Prototipo de WorkFlow Social – SocialFlow .....	103
5.4. CONCLUSIONES DEL CAPÍTULO .....	109
<b>CASO DE ESTUDIO.....</b>	<b>110</b>
6.1. PLANTEO DEL PROBLEMA.....	110
6.2. ETAPAS 1 A 3 – PLAN, REQUISITOS Y MODELADO DEL NEGOCIO .....	111
6.3. ETAPA 4 - MODELADO DE PROCESOS.....	112
6.3.1. Modelado de los Subprocesos.....	112
6.4. ETAPA 5 - MODELADO DE SERVICIOS (P2S). .....	116
6.5. ETAPA 6 - DEFINICIÓN DE COMPONENTES. ....	121
6.6. INCORPORANDO ASPECTOS DE DISTRIBUCIÓN Y SOCIALES.....	123
6.6.1. Aspectos Distribuidos .....	123
6.6.2. Aspectos Sociales .....	125
6.7. COMPARACIÓN Y BENEFICIOS DE LA SOLUCIÓN .....	127
6.8. CONCLUSIONES DEL CAPÍTULO .....	130
<b>CONCLUSIONES.....</b>	<b>131</b>
7.1. RESULTADOS OBTENIDOS .....	131
7.2. CONTRIBUCIONES .....	133
7.3. TRABAJOS FUTUROS .....	136
<b>REFERENCIAS.....</b>	<b>139</b>

# ÍNDICE DE FIGURAS

Figura 1. Ciclo de Vida de los Procesos de Negocio .....	16
Figura 2. Modelo Conceptual de Procesos de Negocio [14].....	18
Figura 3. Elementos de Computación en la Nube.....	27
Figura 4. Modelo de Nube. Visión Integrada .....	30
Figura 5. Modelo de Nube [NIST] .....	30
Figura 6. Modelo OITP .....	33
Figura 7. Modelo OITP con Aspecto Tecno-Social.....	34
Figura 8. Esquema PAD de Distribución [29].....	41
Figura 9. Tres Tipos de Modelos de Despliegue de Servicios .....	56
Figura 10. Etapas de MISP con Interacción Mejorada .....	60
Figura 11. Ortogonalidad de Procesos y Servicios.....	61
Figura 12. Símbolo de Servicio como Círculo con Cuerda.....	64
Figura 13. Marco Metodológico Propuesto: Etapas en Cada Ciclo de Vida.....	68
Figura 14. Marco Metodológico Propuesto: Interacción de Etapas .....	68
Figura 15. Metamodelo Simplificado de BPMN .....	69
Figura 16. Metamodelo de Conceptualización de Servicios SOAF .....	71
Figura 17. P2S Metamodelo de Conceptualización de servicios .....	71
Figura 18. Metamodelo Simplificado de SCA .....	73
Figura 19. Enlaces y Bindings SCA [115].....	74
Figura 20. Vinculación del Metamodelo P2S con SCA .....	75
Figura 21. Metamodelo WSDL 2.0 .....	78
Figura 22. Integración entre los Metamodelos P2S, SCA y WSDL.....	78
Figura 23. Proceso de una Orden de Trabajo .....	81
Figura 24. Servicios del Proceso de OT .....	82
Figura 25. Exportando la OT desde BPMN a P2S.....	83
Figura 26. Prototipo de un Editor Gráfico de Servicios.....	84
Figura 27. Modelo de Componentes SCA para el Ejemplo .....	85
Figura 28. Escenarios Posibles .....	89
Figura 29. BPMS como IaaS .....	90
Figura 30. BPMS como PaaS .....	91
Figura 31. BPMS como SaaS.....	91
Figura 32. Asistente para la Creación del Conector en Bonita.....	95
Figura 33. Arquitectura de la Aplicación de Monitoreo e Interacción del Usuario .....	98
Figura 34. Diagrama de Comunicación de la Aplicación .....	99
Figura 35. Visualización de las Instancias y Actividades .....	99
Figura 36. Visualización en GraphViz .....	100
Figura 37. Arquitectura y Componentes Tecnológicas de SocialFlow .....	105
Figura 38. Formulario Asociado a una Tarea en SocialFlow .....	106
Figura 39. Creación y Edición de Comentarios en SocialFlow.....	107
Figura 40. Mecanismo de Notificaciones en SocialFlow .....	107
Figura 41. Visualización de Notificaciones en SocialFlow.....	108
Figura 42. Espacio de Trabajo del Usuario de SocialFlow .....	109
Figura 43. Subprocesos y Actores del Proceso TesisFlow .....	113
Figura 44. Subproceso Propuesta de Tesina.....	113
Figura 45. Subproceso Desarrollo de Tesina .....	113
Figura 46. Subproceso Validación de Datos Académicos .....	114
Figura 47. Subproceso Análisis de la Propuesta.....	114
Figura 48. Subproceso Presentación de Informe Final .....	115
Figura 49. Subproceso Evaluación de Informe .....	115
Figura 50. Subproceso Coordinación de Fecha de Exposición .....	115
Figura 51. Subproceso Exposición de Tesina .....	116
Figura 52. Subproceso Validación de Datos Académicos .....	119
Figura 53. ProcessService SIU-Guarani .....	119
Figura 54. ProcessService Coordinación Tesinas.....	120
Figura 55. ProcessService Administración y Gestión .....	121
Figura 56. Modelo de Componentes del ProcessService SIU-Guarani.....	121
Figura 57. Componentes SCA del ProcessService Coordinación de Tesinas .....	122
Figura 58. Componentes SCA del ProcessService Administración y Gestión.....	122
Figura 59. Modelo de Interacción de Componentes.....	123
Figura 60. Descomposición y Distribución de TesisFlow .....	125
Figura 61. Subprocesos de TesisFlow en el Modelo de Componentes.....	125

---

Figura 62. Subproceso Evaluación del Informe-Comentarios y Notificaciones .....	127
Figura 63. Subproceso Desarrollo de Tesina con Etiquetas y Adjuntos .....	128

## ÍNDICE DE TABLAS

Tabla 1. Comparación de Conceptos de Orquestación y Coreografía.....	5
Tabla 2. Comparación de Definiciones de Computación en la Nube .....	26
Tabla 3. Comparación de Herramientas y Metodologías.....	67
Tabla 4. Reglas de Integración entre Procesos y Servicios .....	76
Tabla 5. Clasificación de Actividades de cada Sub-Proceso .....	117
Tabla 6. Distribución de Procesos de TesinaFlow .....	124
Tabla 7. Comparación de Funcionalidades .....	129



# ÍNDICE DE DEFINICIONES

Definición 1. PROCESO DE NEGOCIO .....	2
Definición 2. GESTIÓN DE PROCESOS DE NEGOCIO .....	2
Definición 3. GESTIÓN DE PROCESOS DE NEGOCIO o BPM .....	14
Definición 4. LENGUAJE META-OBJECT FACILITY o MOF .....	24
Definición 5. COMPUTACION EN LA NUBE [NIST] .....	25
Definición 6. COMPUTACION EN LA NUBE [ENISA] .....	25
Definición 7. COMPUTACION EN LA NUBE [FURHT ET AL.].....	25
Definición 8. COMPUTACION EN LA NUBE [UCBerkeleyRADLabs.].....	25
Definición 9. COMPUTACION EN LA NUBE [McKinsey] .....	26
Definición 10. BPAAS – PROCESO COMO SERVICIO .....	88
Definición 11. PLATAFORMA EN LA NUBE DE PROCESOS DE NEGOCIO .....	89

# CAPÍTULO 1

## INTRODUCCIÓN

La gestión de procesos de negocio se basa en la idea de que cada producto es el resultado de un conjunto de actividades que se realizan a fin de obtener dicho producto. Por este motivo, la correcta y eficiente gestión de los procesos de negocio es un aspecto importante para la productividad de toda organización, ya que permite identificar las tareas que la organización debe realizar para producir sus productos, el orden de ejecución de las mismas y las personas responsables de realizarlas.

En particular, el enfoque de solución de problemas orientado a la identificación, modelado, despliegue y seguimiento de procesos de negocio, ha permitido que las organizaciones expliciten el conocimiento subyacente en sus operaciones y se involucren en un ciclo de mejora continua.

Las tecnologías de la información y las comunicaciones (TICs) en general y los sistemas de información en particular, juegan un rol importante en la gestión de procesos de negocio, dado que las actividades de dichos procesos constituyen partes de sistemas de información automatizados por medio de las TICs. Por este motivo, el estudio y desarrollo de herramientas TIC para soportar la ejecución de procesos de negocio resulta relevante.

Este capítulo esboza el área principal de la tesis - los procesos de negocio y el ciclo de vida de los mismos mejorado a través de una metodología formalizada. Entre otras mejoras, la metodología propuesta considera el despliegue de los procesos de negocio como modelos ejecutables enriquecidos, facilitando la colaboración entre los actores responsables de la ejecución de las distintas actividades del proceso, así como también incorpora interacciones que comúnmente se llevan a cabo por fuera de los procesos de negocio, si bien forman parte de los mismos.

El capítulo se organiza de la siguiente manera. La Sección 1.1 esboza los conceptos principales en torno a los procesos de negocio, revisando particularidades de su visión clásica y planteando nuevas tendencias. La Sección 1.2 presenta nociones de procesos de negocio sociales y colaborativos. La Sección 1.3 enuncia el problema a resolver, esto es, las mejoras a la Metodología Integradora de Servicios y Procesos (MISP) y la incorporación de mecanismos de distribución y de colaboración que mejoran los rastros de ejecución de los procesos de negocio y favorecen a una mayor interacción entre los involucrados. La Sección 1.4 aborda el enfoque de solución y explica los componentes de la solución propuesta. La Sección 1.5 presenta las principales contribuciones de esta tesis y finalmente la Sección 1.6 describe su estructura.

## 1.1. PROCESOS DE NEGOCIO

El concepto de proceso de negocio representa la ejecución coordinada de actividades en una organización. A continuación presentamos una definición de proceso de negocios y explicamos la relevancia de la gestión de los procesos de negocios en las organizaciones.

### 1.1.1. Definición de Conceptos

Varias definiciones de proceso de negocio y de gestión de procesos de negocio existen en la literatura. Por ejemplo, según Mathias Weske [14]:

#### **Definición 1. PROCESO DE NEGOCIO**

*Un Proceso de Negocio es un conjunto de actividades que se realizan en coordinación en un ambiente organizacional y técnico. Estas actividades alcanzan el objetivo del negocio.*

Un proceso es un ordenamiento específico de actividades a través del tiempo y lugar, con un principio y un final y con entradas y salidas claramente especificadas. Los elementos que componen un proceso son: 1) diagrama o representación gráfica del proceso, 2) actores 3) reglas de negocio y 4) actividades y sub-procesos.

El diagrama de un procesos contiene actividades, eventos, secuencias o bifurcaciones de flujo, puntos de decisión y actividades. Las actividades constituyen los pasos manuales o automatizados de los procesos y son realizadas por actores o participantes que deben seguir el flujo de las mismas.

Las reglas de negocio se establecen para determinar cómo opera la organización para cumplir sus objetivos, se aplican en varios pasos de un proceso y es preciso definir las en forma separada del proceso en sí mismo.

Debido a la importancia de los procesos de negocio en las organizaciones, se hace imprescindible la eficiente administración (o gestión) de los mismos. A continuación proveemos la definición de gestión de procesos de negocios provista por Weske [14].

#### **Definición 2. GESTIÓN DE PROCESOS DE NEGOCIO**

*La Gestión de Procesos de Negocio (o Business Process Management - BPM) incluye conceptos, métodos y técnicas para soportar el diseño, administración, configuración, representación y análisis de los procesos de negocio. Su objetivo último es representar el proceso de negocio con sus actividades y las restricciones de ejecución entre ellas.*



El dominio de la gestión de procesos de negocio resulta de interés para las comunidades de administración del negocio formadas por los analistas de negocio, y de Tecnologías de la Información (TI) integrada por expertos informáticos. Recientemente, BPM ha adquirido una atención considerable tanto por parte de las comunidades de administración de negocios como por la de TI, en particular, por especialistas en Ciencias de la Computación. Sin embargo, en referencia a los procesos de negocio, los miembros de estas comunidades no se ven como pares al momento de colaborar en la implementación de los mismos. Los analistas de negocios tienden a considerar a los expertos informáticos como subordinados a los aspectos del negocio. Mientras que los expertos en tecnología consideran que los objetivos de negocio y las regulaciones organizacionales no merecen mucho análisis ni requieren alto nivel de abstracción para la implementación de las soluciones de TI.

Uno de los principales objetivos que persigue el enfoque de resolución de problemas orientado a procesos de negocio, es conciliar los puntos de vista de los analistas de negocios y de los expertos informáticos y lograr modelizar e implementar procesos de negocio correctos y robustos en un software adecuado, contribuyendo a la mejora continua de los procesos y a una ágil absorción de los cambios de la realidad [1]. El éxito en el alcance de los objetivos y que estos se alcancen de manera eficaz y eficiente depende del trabajo coordinado de los recursos que intervienen, ya sean estos humanos o tecnológicos.

En el ámbito organizacional, los procesos de negocio sirven para entender cómo opera la institución y juegan un rol importante en la construcción de sistemas de información flexibles. Esta flexibilidad se mide en la capacidad de adaptarse rápidamente a los cambios para absorber velozmente el ritmo del mercado.

Un aspecto importante de los procesos de negocios es que los mismos atraviesan y trascienden las estructuras organizativas. Usualmente, un proceso de negocio incluye actividades que se realizan a través de varias unidades organizativas de la empresa. La responsabilidad de ejecutar cada actividad es asignada a una persona o área, incluso si la actividad es automatizada. De esta manera, las actividades de los procesos representan el trabajo a realizar por cada área, y a través de la correcta ejecución de las mismas se puede controlar que cada área cumpla con la misión que tiene encomendada. Es por esto que el enfoque orientado a procesos de negocio implica un fuerte énfasis en cómo el trabajo es realizado dentro de una organización o entre organizaciones. Este enfoque se contrasta con el enfoque orientado a producto, el cual enfatiza lo que cada área organizacional produce.

Las políticas empresariales y las reglas de negocio se establecen para determinar cómo debe actuar la empresa para cumplir sus objetivos, respondiendo a estrategias preestablecidas. Estas reglas son de aplicación en los distintos pasos de los procesos de negocio de la organización. A fin de realizar un seguimiento efectivo de como las políticas empresariales y las reglas de negocio se implementan en la práctica es necesario gestionar los procesos de negocio. Por este motivo, la definición de un proceso conlleva a identificar indicadores que permitan medir su rendimiento, favoreciendo el análisis y la

mejora continua de dicho proceso. Asimismo, la gestión de procesos de negocios se torna una tarea relevante para administrar una organización.

Los objetivos de la gestión de los procesos de negocio pueden resumirse en los siguientes puntos:

Explicitar los procesos de negocio para entender las operaciones realizadas en una organización y las relaciones entre dichas operaciones.

Identificar actividades y representarlas en un modelo de procesos de negocio, a fin de facilitar la comunicación entre los participantes, permitiendo el análisis riguroso del modelo así como también su mejora.

Facilitar la mejora continua de los procesos de negocio, cuyo repositorio constituye un activo para la organización y su gestión.

Reducir la brecha existente entre el proceso de negocio en sí mismo y su realización en un sistema de software.

### **1.1.2. Orquestación y Coreografía**

La implementación de un proceso de negocios, cuando es automatizada, es facilitada por servicios de software - piezas de software que implementan una determinada funcionalidad. En la práctica, existe una relación de "realización" entre procesos de negocio y servicios de software. Por un lado, los procesos de negocio atraviesan la estructura organizativa y sus actividades satisfacen las reglas de negocio. Más aún, la orientación a procesos implica independizarse de la estructura organizativa, y pensar las actividades según la manera en que se ejecutan en lugar de dónde se realizan. Por otro lado, los servicios de software resuelven funcionalidades concretas requeridas dentro de cada unidad organizativa y se pueden componer para realizar procesos de negocio.

En la relación descrita previamente, los procesos de negocios se comportan como consumidores de los servicios de software (servicios). Así, mientras los procesos de negocio atraviesan la estructura organizativa y definen sus reglas independientemente del proceso de desarrollo, los servicios resuelven funcionalidades concretas requeridas dentro de cada unidad organizativa y se componen para realizar los procesos de negocio. La composición de servicios para la realización de un proceso de negocios se puede realizar a través de su orquestación y coreografía [2]. Estos dos conceptos se explican a continuación.

Muchas organizaciones plasman en un Sistema de Gestión de Procesos de Negocio – software que convierte el modelo de procesos en una versión ejecutable, en inglés denominado Business Process Management Systems (BPMS) - todas sus actividades como un componente de software centralizado. Este control centralizado es conocido como *orquestación* [10]. La orquestación se trata de un enfoque para componer servicios y procesos de negocio donde se define la secuencia de pasos en un proceso, se establecen las condiciones excepcionales y luego se fija un controlador central para ejecutar la secuencia.

A diferencia de la orquestación de servicios, la interacción de servicios a fin de satisfacer la funcionalidad requerida por un proceso de negocios se denomina

*coreografía* [9]. Este término indica la ausencia de un agente central que controle las actividades involucradas en el proceso. La interacción entre servicios solamente se basa en el envío y recepción de mensajes. Se trata de un enfoque donde las reglas se definen para cada participante del proceso y el comportamiento general se basa en la interacción de las piezas individuales.

En la tabla 1 se presenta una comparación entre ambos conceptos – Orquestación y Coreografía, utilizando como criterio los siguientes elementos: el objetivo de cada enfoque, el modelo o metáfora que siguen, el enfoque que adoptan y el fundamento para su uso.

**Tabla 1. Comparación de Conceptos de Orquestación y Coreografía**

CRITERIO	ORQUESTACIÓN	COREOGRAFÍA
Objetivo	Componer servicios para cumplir con un proceso de negocio dentro de una organización	Componer servicios para colaboración entre unidades organizaciones
Modelo	Jerárquico Pregunta-Respuesta	Peer-to-Peer
Enfoque	Componer servicios y definir el orden en que son ejecutados para alcanzar el objetivo de un proceso de negocio	Definir la manera en que múltiples partes colaboran para conformar una transacción de negocio
Fundamento	Constituye un servicio en sí mismo	Define la interacción del negocio

Es importante tener en cuenta que una buena resolución de procesos de negocios garantiza una buena solución orientada a servicios y no viceversa. La noción de una aplicación o servicio compuesto se basa en la idea de la construcción de nuevas aplicaciones o servicios, interconectando las partes existentes. La orquestación juega un papel importante en esto, ya que es quien aglutina estas partes al coordinar la ejecución de cada servicio discreto de manera centralizada. Es por esto que en la orquestación debe existir un mecanismo que dirija las actividades. Estas actividades son en realidad interacciones entre servicios, es decir, servicios que se invocan unos a otros, pero no de forma desordenada, sino de manera controlada por el orquestador que es quien conoce el detalle de todas las tareas que se deben llevar a cabo para completar el proceso. La construcción del proceso de negocio se realiza en dos pasos: primero se publican los servicios y luego se orquestan, es decir, se integra cada servicio al proceso en su lugar y momento adecuado.

En la orquestación de servicios hay varios actores involucrados. Entre ellos encontramos la especificación del proceso de negocio, un motor de ejecución de procesos que contiene los procesos de negocio y sus reglas, y los consumidores de los servicios que se exponen.

A diferencia de la orquestación, la coreografía plantea un esquema en donde no hay un control centralizado del proceso, sino un control “declarativo” que sólo especifica cuáles son las interacciones permitidas entre dos pares. De esta forma, dadas las reglas correctas, las partes interactuarán unas con otras en un estilo “peer-to-peer” y el proceso de negocio estará definido de forma implícita.

De ahí su nombre (coreografía), ya que se asemeja a un estilo en donde cada parte hace su trabajo bajo ciertas reglas y se obtiene un resultado final conjunto.

Estos dos conceptos de orquestación y coreografía están íntimamente vinculados a la ejecución de los procesos más que a su modelado. Como se detallará en las próximas secciones, el aspecto de la ejecución de los procesos es uno de los que se pone en discusión en torno a su capacidad de absorber nuevos paradigmas. Pero sin lugar a dudas tanto la orquestación como la coreografía mantienen su vigencia al investigar temas de procesos y de negocios y además se transforman en protagonistas importantes a la hora de dar respuestas a nuevas maneras de transitar el ciclo de vida de los procesos de los procesos de negocio.

### 1.1.3. Nuevas Tendencias

Los procesos de negocio conceptualizados de la manera clásica resultan útiles para el modelado y ejecución de tareas repetitivas y estructuradas. Esta característica es consecuencia de los elementos notacionales utilizados para su modelado pero también responde a un modelo organizacional rígido donde los flujos de ejecución no cambian y la coordinación entre las actividades es bien ordenada. Esta concepción de los procesos de negocio se torna insuficiente cuando los cambios en la realidad afecta el modo en que dichos procesos se ejecutan y también cuando la interacción entre los participantes no queda incluida en la ejecución del proceso y, en consecuencia, no deja rastros.

En base a esas consideraciones, se han identificados características como *model-reality divide* [3] que describen la división entre los modelos de procesos abstractos y los ejecutados en la realidad. Los trabajos actuales revelan nuevos puntos de vistas o perspectivas que han ejercido gran influencia sobre el diseño de notaciones y lenguajes para procesos de negocio, tanto en el modelado como en la ejecución.

Otro aspecto novedoso en estas nuevas tendencias, es la clasificación de los procesos de negocio en dos categorías: aquellos procesos de negocio bien definidos y repetitivos con necesidades de una coordinación y automatización bien claras, y los procesos de negocio menos estructurados (también denominados dinámicos o *knowledge-intensive*). Estos últimos tipos de procesos se caracterizan por la necesidad de compartir información entre los actores más que por la necesidad de coordinar sus actividades.

Por otra parte, el BPM clásico es un estándar donde existe una pérdida de innovación [4]. Si bien hay conocimiento de la organización y se explicita el mismo a través de modelos legibles, existe una gran cantidad de conocimiento que se despliega en tiempo de ejecución que no solamente no es registrado ni considerado, sino que favorece la desactualización de los modelos e impide contribuir a una verdadera mejora del proceso.

En la próxima sección se presentan los conceptos relacionados a lo que se denominan procesos de negocio sociales – conocido como Social Business

Process Management o *Social BPM*) y se incorpora también el paradigma de colaboración, intrínseco de los entornos sociales, a los procesos de negocio.

## 1.2. PROCESOS DE NEGOCIO SOCIALES Y COLABORATIVOS

Uno de los aspectos de interés planteados como trabajo futuro durante las tareas de investigación que contribuyeron a la definición de MISP [1] ha sido el estudiar la aplicación de BPM al concepto de redes sociales con Web 2.0. Esta tesis, retoma y desarrolla ese aspecto y cuestiones relacionadas.

En [5], se define la Web 2.0 como una segunda generación de comunidades basadas en la Web y como una alternativa al hospedaje de servicios (tales como redes sociales y wikis) que facilitan la creatividad, colaboración y mecanismos de intercambio entre usuarios.

Por otra parte, en [6] se plantea que las redes sociales se han transformado en uno de los fenómenos tecnológicos más dinámicos facilitados por Internet. Si bien comenzó como un fenómeno meramente social, las organizaciones ya están vislumbrando a las redes sociales como una herramienta poderosa para compartir información y favorecer el trabajo en grupo.

En este sentido, en [6] se hace hincapié en la presencia de las “3C” - Colaboración, Comunicación y Coordinación; del concepto de *groupware*, pero en una red social. El autor afirma que las redes sociales tradicionales carecen del aspecto de “coordinación” siendo BMP y los workflow mecanismos válidos para aportar este aspecto a las redes sociales.

Bajo estas premisas, es válido tomar de ambos paradigmas los conceptos básicos que conduzcan al aprovechamiento de cada uno de ellos y que además reflejen fielmente la realidad de las organizaciones actuales que observan en estos dos enfoques, las respuestas que permitan abordar flexiblemente y con la mayor información posible los cambios que impone en entorno, sea este el mercado, en caso de organizaciones privadas, o las regulaciones y normas en el caso de las organizaciones públicas.

### 1.2.1. Debilidades de BPM Respecto de los Conceptos de Software Social

Como se plantea en la sección 1.1.3 desde una concepción social, el concepto de *model-reality divide*, es una de las principales debilidades de BPM. Esta división en el modelado de la realidad deja afuera a los usuarios finales, que son los que en verdad quienes realizan la ejecución de los procesos y advierten si los requisitos son cubiertos parcialmente o no cubiertos. Una vez más, la dificultad está puesta en el hecho de que los modelos no se condicen con las versiones ejecutables y, peor aún, no se retroalimentan entre ellos.

Siempre revisando el BPM clásico y dentro de su ciclo de vida clásico, cabe analizar las etapas de monitoreo y mejora continua de los procesos de negocio. En este aspecto resulta también de gran valor el aporte de los usuarios finales ya que son los últimos responsables de la ejecución de los procesos y conocen

en detalle sus debilidades. El software social integrado a BPM permitiría tomar en cuenta los aportes de estos actores, como así también incorporar al proceso conocimiento no estructurado y aplicarlo a la mejora del mismo.

En este mismo orden de cosas, siempre analizando el BPM clásico, las particularidades estrictas acerca de cómo los usuarios acceden a los procesos y los ejecutan, junto con la imposibilidad de incorporar el contexto de ejecución, perjudican tanto la incorporación de conocimiento como también la posibilidad de integrar actividades, muchas veces no previstas en los procesos, y que forman parte del mismo. De este modo, la única alternativa posible es diseminar información y facilitar flujos por fuera del proceso en ejecución.

Tal como se afirma en [3], es evidente que los procesos de negocio *knowledge-intensive* actuales no pueden estar representados por flujos de control rígidos y requieren nuevas perspectivas para entender y compartir su complejidad entre los involucrados.

### 1.2.2. Software Social y BPM Social

El software social es un término acuñado para definir al conjunto de herramientas y aplicaciones que habilitan la interacción grupal y la comunicación mediada por un computadora [7][8]. Es decir provee una plataforma de colaboración y comunicación entre individuos y grupos a través de uso de TICs.

Las claves principales del software social han sido enunciadas por varios autores [3][8] e incluyen:

- **Lazos débiles** – definidos en contraposición a los lazos fuertes y jerárquicos de las estructuras organizacionales y donde se definen los procesos de negocio clásicos. Si bien los procesos de negocio son transversales y eso contribuye a una visión integral de la organización, no han sido capaces aún de aplanar la pirámide de las organizaciones y desestructurar sus interacciones.
- **Producción social** – el favorecimiento de contribuciones innovadoras e inesperadas, como por ejemplo la retroalimentación de un usuario, sin tener en cuenta su ubicación geográfica. En este sentido, el software social posee una característica global absolutamente inexplorada en los procesos de negocio, cuyo alcance es la organización y sus fronteras.
- **Igualitarismo** – en el software social se incrementa el rol de la confianza entre los usuarios dentro de la organización, de modo que todos contribuyan con los mismos derechos. Este aspecto está muy vinculado a la característica de lazos débiles ya mencionada.
- **Provisión de servicios mutuos** – el software social transforma el modelo organizacional de un mecanismo unidireccional en la prestación de servicios a un sistema de intercambio. En este sentido los conceptos de orientación a

servicios presentes en el paradigma BPM y largamente revisados en [1], mantienen su vigencia.

En base a las características anteriores, se puede observar que el software social cumple con dos de las premisas requeridas por el concepto 3C explicado en la sección 1.2 – comunicación y colaboración. La complementación de la tercera C - coordinación, puede ser aportada por BPM Social y sistemas de workflow, como propone esta tesis. Es evidente que tanto los BPM Social como el software social abordan la gestión de actividades de trabajo, aunque considerando diferentes fuentes de información. Sin embargo, como se menciona en [4], el software social provee nuevas herramientas a considerar:

- **Auto-identificación** – cualquier actor puede contribuir a una actividad y llevarla a cabo.
- **Transparencia** – los resultados del flujo de trabajo deben estar disponibles para todos.
- **Firma** – la intervención de cada actor debe estar firmada al momento de completar cada actividad.
- **Modificación abierta** – cualquiera puede modificar las contribuciones de otros actores.
- **Autenticación o logging** – todas las actividades deben estar autenticadas para conocer la historia de su ejecución.
- **Discusión** – los comentarios y sugerencias sobre los resultados del trabajo deben estar vinculados a las actividades realizadas.
- **Poder de rechazo** – los actores que exhiban comportamientos inapropiados pueden ser prohibidos.

Estas características propias del software social, vistas a la luz de BPM pueden resultar contrapuestas, sin embargo, optimizando y mejorando las tecnologías y metodologías del BPM tradicional se alcanza el concepto de Social BPM. Si bien se trata de un tema que se encuentra en etapas de investigación muy tempranas, no caben dudas que incorporar estas características a la ejecución tradicional de los procesos de negocio resulta beneficiosas y complementarias para obtener mejores resultados para las organizaciones.

Habiendo introducido la problemática actual en torno a la definición de métodos y técnicas mejoradas en el área de la gestión de procesos de negocio, en la próxima sección enunciamos el problema abordado en esta tesis.

### 1.3. ENUNCIADO DEL PROBLEMA

Los procesos de negocio en la realidad actual no solamente atraviesan la organización sino que deben adaptarse para considerar dos aspectos con los que conviven todas las organizaciones. Por un lado, y como se ha planteado en la sección anterior, la colaboración entre los participantes no sólo para responder a las actividades que se le han asignado sino para producir aportes significativos que mejoren los resultados y que además se vean reflejados en los modelos. Por otra parte, independientemente de si ha sido impuesto por el mercado o por la tecnología, las organizaciones actuales poseen una globalidad que las hace independientes de la ubicación geográfica y donde la distribución es moneda corriente. Los procesos de negocio no están ajenos a esta realidad, siendo uno de los directamente afectados dada la transversalidad que modelan.

Así como las organizaciones necesitan resolver problemas cada vez más complejos, surgen nuevas soluciones tecnológicas que permiten dar respuestas a estos problemas. Por ejemplo, el concepto de computación en la nube (*cloud computing*) aporta un nivel superior de eficiencia para distribuir y desplegar recursos de TI bajo demanda. La incorporación de conceptos como virtualización, despliegue bajo demanda, distribución de recursos por Internet y pago por uso cambian el enfoque tanto de los modelos de procesos de negocio como de la ejecución de dichos procesos.

Los problemas de investigación abordados en esta tesis buscan mejorar y actualizar la Metodología Integradora de Servicios y Procesos (MISP) propuesta en [1]. Las mejoras y actualizaciones tienen como objetivo obtener una solución que siga teniendo en cuenta a la organización como un todo, pero que facilite la revisión de los aspectos de los modelos de los procesos de negocio que la representan, y de los servicios de software que ejecutan dichos procesos. Las mejoras se basan en almacenar y utilizar rastros en la ejecución de los procesos de negocio a fin de retroalimentar la mejora continua de los mismos.

En esencia, la actualización que se propone para MISP puede analizarse desde dos puntos de vista: 1) mejoras vinculadas al modelado de procesos y servicios, como así también al mecanismo de interacción entre procesos y servicios – estas mejoras están relacionadas con aspectos conceptuales y metodológicos de la propuesta; y 2) mejoras a la ejecución y despliegue de los procesos de negocio – las mismas revisan aspectos tecnológicos y analizan la incorporación de nuevos enfoques – en particular, software social y computación en la nube.

#### 1.3.1. Mejoras Vinculadas al Modelado de Procesos y Servicios

Las mejoras propuestas a MISP en torno a los modelos que representan cada etapa, se enfocan en mejorar la interacción entre los procesos de negocio y los servicios de software, formalizando cada fase de la metodología para permitir la generación de productos de software más robustos y confiables y ampliar los mecanismos para explicitar el conocimiento intrínseco en los procesos y servicios. En este sentido, se propone utilizar metamodelos como medio para



definir la sintaxis de los modelos utilizados en cada etapa de la metodología así como proponer reglas de transformación entre cada etapa.

La propuesta se basa en definir nuevos metamodelos o vincular metamodelos ya definidos. En particular, para reducir la brecha entre el modelado de procesos y el de servicios, se propone un metamodelo de conceptualización de servicios denominado P2S y su vinculación al metamodelo de BPMN [11]. Adicionalmente, para la etapa de definición de componentes de MISP, se integra P2S con un metamodelo de componentes [12] definido por el estándar Service Component Architecture (SCA). Finalmente, se combinan dos estándares bien conocidos para la interacción entre la definición de componentes y su implementación como servicios [13], integrando SCA con Web Services Definition Language (WSDL).

### **1.3.2. Mejoras Vinculadas a la Ejecución y Despliegue**

Las mejoras propuestas en relación con aspectos de ejecución y despliegue no afectan en forma directa a la metodología pero definen cuestiones vinculadas con la elección o construcción de herramientas de software que transformen los modelos de procesos de negocio y servicios en versiones ejecutables (BPMS). En este sentido, este trabajo revisa dos aspectos: 1) la distribución de procesos, como mecanismo para mejorar la capacidad de escalar la complejidad de los mismos y también como la manera de utilizar y beneficiarse de nuevos enfoques de TI, como la computación en la nube, considerado como un nuevo modelo de servicios que afecta a toda la industria del software; y 2) la incorporación de aspectos colaborativos durante la ejecución de los procesos de negocios a fin de mejorar la participación de los actores y enriquecer los rastros de ejecución que retroalimentan el ciclo de mejora continua e, indirectamente, mejoran los modelos producidos.

## **1.4. ENFOQUE DE SOLUCIÓN**

La solución al problema planteado se basa en las siguientes tareas:

- 1) Revisar y analizar las distintas etapas que conforman MISP, fundamentalmente en torno a mejorar los mecanismos de interacción entre las etapas, así como proveer una definición mejorada de las mismas.
- 2) Incorporar los conceptos de colaboración y distribución en toda la metodología y aplicarlos al ciclo de vida de los procesos de negocio para perfilar una posible arquitectura de Social BPM.
- 3) Analizar las características globales actuales de las organizaciones y de nuevas tendencias en software que las organizaciones utilizan para que las mismas sean consideradas por el marco metodológico planteado.

Como se mencionó anteriormente, las mejoras a los mecanismos de interacción entre etapas de la metodología se abordan desde el enfoque de los metamodelos y la definición de reglas de transformación. Este enfoque mejora y unifica las sintaxis para definir y especificar servicios a partir de las actividades

de los procesos y plantea reglas de transformación hacia metamodelos de estándares para definición de componentes y servicios web.

Respecto de la incorporación de conceptos de colaboración y distribución en procesos de negocios, los mismos se incluyen proponiendo un modelo arquitectónico de distribución que considere aspectos relevantes relacionados a los procesos - como por ejemplo, la frecuencia de acceso a los datos, la disponibilidad del motor de procesos de negocio, y la sensibilidad de los datos.

Finalmente, en cuanto a la inclusión de aspectos de software social, se propone un prototipo de BPM Social denominado SocialFlow que incluye funcionalidades que permiten enriquecer la colaboración entre los participantes y deja rastros de ejecución mejorados. Este prototipo se basa en una especificación de requisitos funcionales derivada de las necesidades identificadas en el software social bajo la visión de los procesos de negocio.

## **1.5. CONTRIBUCIÓN DE LA TESIS**

La principal contribución de este trabajo es un marco metodológico y una arquitectura tecnológica para abordar el modelado y ejecución de procesos de negocio integrados con servicios, en un contexto distribuido y con características sociales y colaborativas.

El marco metodológico propuesto constituye una mejora a la metodología MISP propuesta en [1] que se enfoca en proponer el modelado de procesos y servicios de manera integrada por metamodelos, con el objeto de mejorar el producto de cada etapa de la metodología y contribuir a un modelado más formal que permita la automatización del desarrollo. Respecto de la arquitectura tecnológica, se analiza en profundidad las innovaciones que pueden incluirse en la fase de ejecución de los procesos de negocio que tiendan a mejorar específicamente el monitoreo de los mismos como elemento fundamental para cerrar el ciclo de vida de los procesos de negocio, contribuyendo a la mejora continua de los mismos. Finalmente, la aplicación de una metodología independiente de las herramientas de TICs, permite aprovechar los resultados de investigación y los avances tecnológicos, a fin de aplicarlos para proponer una nueva arquitectura tecnológica.

## **1.6. ESTRUCTURA DE LA TESIS**

Este trabajo se estructura de la siguiente manera. A continuación de esta introducción, en el Capítulo 2 se introducen los conceptos principales de vinculados a BPM, computación en la nube, redes sociales y aspectos relacionados. En el Capítulo 3 se analiza la bibliografía considerada para la elaboración de este trabajo mostrando el estado del arte en materia de BPM y sus tecnologías y herramientas actuales en torno a los conceptos analizados en el capítulo 2. En el Capítulo 4 se presenta una primera mejora a la metodología integradora de servicios y procesos (MISP) en torno a la interacción de etapas y

la adecuación de modelos. En el Capítulo 5, se completa la ampliación del marco metodológico en torno a las características distribuidas y colaborativas de procesos y servicios. En el Capítulo 6 se valida la propuesta a través de un caso de estudio. Finalmente, en el Capítulo 7 se presentan conclusiones, se explican las contribuciones de la tesis y se discuten posibles trabajos futuros.

## CAPÍTULO 2

### CONCEPTOS RELACIONADOS

En este capítulo se describen los conceptos relacionados a los temas abordados en la tesis. En la Sección 2.1 se introducen conceptos relacionados a Gestión de Procesos de Negocios, en inglés conocido como Business Process Management (BPM). En la Sección 2.2 se presentan las particularidades del Desarrollo de Software Dirigido, en inglés Model Driven Development (MDD). La Sección 2.3 detalla definiciones y escenarios tecnológicos de la computación en la nube, así como la Sección 2.4 describe software social. Por último en las Secciones 2.5 y 2.6 se analizan las características de BPM respecto de la computación en la nube y del software social.

#### 2.1. GESTIÓN DE PROCESOS DE NEGOCIOS

En esta sección se presentan conceptos básicos de la gestión de procesos de negocio (BPM) con el objetivo de introducir al lector en una de las áreas que sustentan la realización de esta tesis.

##### 2.1.1. Definiciones

Existen en la literatura diversas definiciones de la gestión de procesos de negocio. Una de ellas, de Weske [14], se introdujo en la Sección 1.1.1.

La fundamentación de la gestión de procesos de negocio se basa en la representación explícita de los procesos de negocio con sus actividades y en el modelamiento y control de las restricciones que pudieran existir en la ejecución de las actividades. Una vez que los procesos de negocio se definen, pueden quedar sujetos a análisis, mejora y promulgación.

Como se explicó en el Capítulo 1, los procesos de negocio necesitan gestionarse. Una definición de BPM, provista por Garimella [15] se presenta a continuación.

##### **Definición 3. GESTIÓN DE PROCESOS DE NEGOCIO o BPM**

*BPM se define como un conjunto de métodos, herramientas y tecnologías usadas para diseñar, promulgar, analizar y controlar los procesos de negocio operacionales.*

BPM trata de un enfoque centrado en procesos para mejorar el funcionamiento organizacional que combina tecnologías de información con procesos y metodologías de gobernanza. BPM es una colaboración entre personas del negocio y del área de TI para promover agilidad, efectividad y transparencia a los procesos de negocio. BPM abarca personas, sistemas, funciones, negocio, clientes, proveedores y socios.

Los Sistemas de Gestión de Procesos de Negocio o BPMS (BPMS - *Business Process Management Systems*) son softwares utilizados como herramientas para representar y coordinar las actividades involucradas en un proceso de negocio. Son los encargados de convertir el modelo de los procesos de negocio en procesos ejecutable.

A la luz de estas definiciones se pueden obtener los puntos clave que resultan de interés conocer como los fundamentos de BPM que aportan a la motivación de este trabajo y que se detallan en las secciones siguientes.

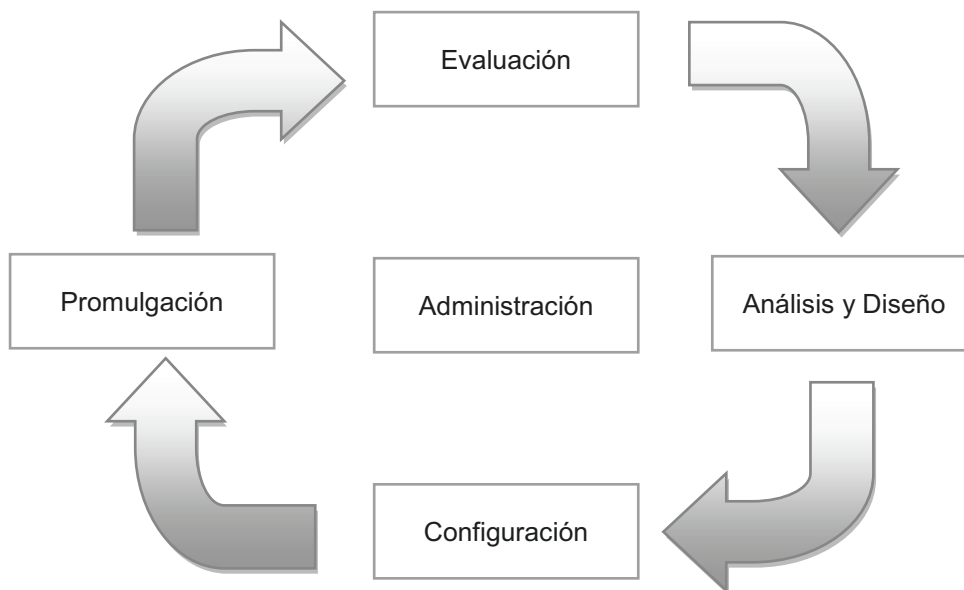
### 2.1.2. Ciclo de Vida de los Procesos de Negocio

En la Definición 2 (Sección 1.1.1) se mencionan las acciones de los procesos de negocio - diseño, administración, configuración, representación y análisis; que se pueden llevar a cabo a través de métodos y técnicas, lo que define el ciclo de vida de los procesos de negocio. Como se expone en [1] el ciclo de vida de los procesos de negocio se compone de fases cíclicas que no implican necesariamente un orden temporal, pero sí una dependencia lógica. Muchas de las actividades de diseño y desarrollo se llevan a cabo dentro de cada fase. Es frecuente también que varias actividades concurrentes se realicen dentro de cada etapa en forma gradual y evolutiva.

En [14] se define un ciclo de vida para los procesos de negocio que consta de cinco etapas como se ilustra en la Figura 1 – Análisis y Diseño, Configuración, Promulgación, Evaluación, y Administración. En la misma se observa claramente un lazo iterativo sin fin donde ninguna de las fases graficadas con rectángulos presenta un orden ni tampoco una condición de salida. La administración como etapa central, sin flujos que la atraviese, pone de manifiesto la presencia de esa fase en cualquiera de las anteriores. Cada una de las etapas se explica a continuación.

- La etapa de **Análisis y Diseño** consiste en el estudio de la situación tanto desde el punto de vista técnico como organizacional. En esta etapa se identifican, revisan, validan y representan los procesos de negocio en un modelo. Sobre la base de este modelo se valida, simula y verifica el proceso, siendo éstas las actividades de análisis.
- En la etapa de **Configuración** se implementa efectivamente el proceso de negocio, y entra en juego la elección o no de un BPMS para dar soporte a la implementación y despliegue de los procesos. En el caso de no utilizar un BPMS las políticas y procedimientos (reglas del negocio) que deban cumplirse, tendrán que ser desarrolladas como un componente más de la solución de software. La

configuración y sobre todo la integración con los sistemas operacionales, es un aspecto de mucha importancia ya que muchos procesos de negocio están soportados actualmente sobre sistemas de software existentes. Por otra parte, la configuración de un BPMS también puede incluir aspectos transaccionales en el sentido estricto de la palabra y el cumplimiento de las propiedades ACID – en inglés Atomicity (atomicidad), Consistency (consistencia), Isolation (aislamiento/independencia), Durability (durabilidad). Si bien, la administración de las propiedades ACID es un tópico resuelto en los sistemas de gestión de bases de datos, no lo es aún en un BPMS, debido a que no contemplan el concepto de unidad atómica de trabajo como constructor propio, sino que debe definirse como parte de la solución que se despliega en el mismo.



**Figura 1. Ciclo de Vida de los Procesos de Negocio**

- La **Promulgación** del proceso abarca el tiempo real de ejecución del proceso y se asocia por analogía con la promulgación de una ley o disposición, que consiste en hacer pública la misma. Esto otorga al proceso de negocio un carácter diferente al del simple despliegue de una pieza de software que implementa un proceso de negocio. Las instancias generadas con cada modelo de un proceso de negocio, se ejecutan de una manera única y repetitiva cumpliendo siempre las mismas restricciones y ejecutando la misma lista de actividades. El BPMS controla y monitorea la ejecución de cada instancia de proceso de negocio. Esta etapa requerirá necesariamente de un BPMS.
- La **Evaluación** de los procesos utiliza la información disponible para evaluar y mejorar los modelos de los procesos de negocios y sus implementaciones y constituye una etapa imprescindible para completar el ciclo de mejora continua de los procesos. Requiere disponer de los registros de ejecución de dichos procesos y la posibilidad de evaluar y simular cambios.
- La etapa de **Administración** es una fase permanente de la gestión de procesos, por lo tanto se visualiza en el centro del ciclo ya que posee una mirada directa a

cada etapa. La complejidad de la administración de procesos variará en función del soporte de software con que se cuente, el número de procesos que se modelen y las características de la organización.

Lo importante de los ciclos de vida de los procesos es facilitar que los participantes cooperen en el diseño y despliegue de una solución que represente el proceso de negocio.

### 2.1.3. Modelado de Procesos - Fundamentos

Los procesos de negocio constan de actividades cuya ejecución coordinada cumple con un objetivo de negocio. Estas actividades pueden ser actividades ejecutadas por un sistema de software, interacciones de usuario o actividades manuales (no soportadas por sistemas automatizados). Las actividades pueden tener una representación textual y ser enumeradas en una lista. Sin embargo, siguiendo este enfoque no se estaría representando el orden según el cual las mismas se llevan a cabo y tampoco se estaría explicitando las restricciones que se aplican a dicha ejecución, no cumpliendo de esta manera con la definición de la gestión de procesos de negocio [14]. En este sentido, las notaciones gráficas constituyen un medio adecuado para expresar el orden de las actividades y si se suma a dicha notación, los símbolos necesarios, también podrán expresarse las restricciones de ejecución.

Un modelo de procesos de negocio es un conjunto de modelos de actividades y las restricciones de ejecución de las mismas. Una instancia de proceso de negocio representa un caso concreto de ejecución del proceso modelado y se compone de instancias de actividades. Más allá de estas definiciones, el término proceso de negocio se utiliza para nombrar indistintamente su modelo o su instancia, ocurriendo lo mismo con el término actividad.

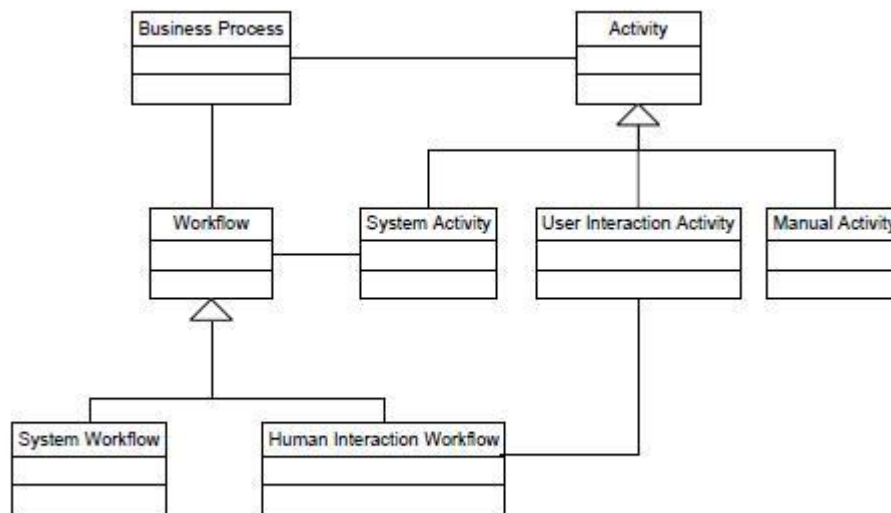
Los modelos de procesos de negocio constituyen el principal artefacto para implementar los mismos. La implementación se realiza a través de sistemas de software (BPMS) que se ajustan a las reglas y políticas organizacionales.

Como se menciona al comienzo de esta sección, se pueden clasificar tres tipos de actividades dentro de los procesos de negocio:

- 1) **Actividad del Sistema** – Las actividades no involucran la interacción humana y son ejecutadas de manera automática por sistemas de software. El usuario sólo provee los parámetros necesarios para la invocación de la actividad por parte del software.
- 2) **Actividad de Interacción del Usuario** – Las actividades con interacción humana son las que ejecutan los participantes del proceso, aplicando su conocimiento pero utilizando sistemas de información – e.g. el ingreso de datos de reclamos de clientes en un entorno de mesa de ayuda o call-center.
- 3) **Actividad Manual** – Las actividades ejecutadas por personas sin mediar la existencia de un sistema de software para llevarlas a cabo. Las actividades

manuales provocan un cambio de estado en el proceso que debe registrarse a través de BPMS.

La Figura 2 muestra mediante un diagrama de clases en Unified Modeling Language (UML) los conceptos y terminologías presentes en el modelado de procesos de negocio y los elementos necesarios para ejecutar los mismos. Como puede observarse, las actividades (*Activity*) son una generalización de los tres tipos de actividades previamente enunciadas – Actividad del Sistema (*System Activity*), Actividad de Interacción del Usuario (*User Interaction Activity*) o una Actividad Manual (*Manual Activity*). Los procesos de negocio (*Business Process*) poseen actividades (*Activity*) y un flujo de trabajo (*Workflow*). Un flujo de trabajo es una generalización de un sistema de workflow (*System Workflow*) – que representa el motor de ejecución de los flujos - y de un flujo de trabajo de interacción humana (*Human Interaction Workflow*).



**Figura 2. Modelo Conceptual de Procesos de Negocio [14]**

#### 2.1.4. Las Tres Dimensiones de BPM

La misma sigla de la gestión de procesos de negocio (BPM) define las tres dimensiones principales de este concepto [15]:

- **Negocio** – (Business, B) es la dimensión del valor de los procesos, ya que es la que considera el valor que el proceso aporta a todos los involucrados. Los objetivos del negocio son alineados con los procesos, que se definen y gestionan aportando la capacidad de mejorar la productividad, incrementar la innovación, elevar los niveles de efectividad del equipo de trabajo e intensificar la lealtad y satisfacción de los destinatarios.
- **Proceso** – (Process, P) es la dimensión de la transformación. La dimensión del proceso crea valor a través de actividades estructuradas. Los procesos



operacionales transforman los recursos y materiales en productos y servicios para los involucrados. Esta transformación es la manera en que el negocio trabaja. Cuanto más efectiva sea esta transformación, mayor valor se crea.

- **Gestión** – (Management, M) es la dimensión de la habilitación. La gestión es la dimensión que hace posible o habilita a los procesos de negocio. La gestión establece personas y sistemas que ponen en ejecución los procesos y persiguen sus objetivos. Para la gestión, los procesos son las herramientas que forjan el éxito del negocio.

Con BPM se unifican sistemas, métodos, herramientas y técnicas en una arquitectura completa que otorga la visibilidad y control necesarios para poder dirigir y afinar los objetivos de los procesos de negocio.

### **2.1.5. Nuevos Desafíos para BPM**

En esta sección se analizan algunas categorías propuestas en [15] como contribuciones prototípicas de BPM. Estas características incluyen: 1) globalización, 2) mercantilización, 3) productividad, 4) innovación, 5) conformidad, y 6) cambios socio-culturales. Todas ellas se toman como base para sustentar las ideas y aportes de este trabajo en cuanto la evolución natural del concepto de BPM para acompañar los avances técnicos y metodológicos a los que las organizaciones se enfrentan actualmente.

#### **Globalización**

Las organizaciones actuales se encuentran en permanente búsqueda para poder sacar cada vez más provecho en términos de costo, calidad e innovación. Estas tres variables están en juego a la hora de medir el éxito mediante la integración de procesos y el intercambio instantáneo de información a nivel global. Esto define un nuevo modelo organizacional global donde las fronteras se diluyen pero al mismo tiempo se deben afrontar diferentes regulaciones, monedas, modalidades operativas, zonas horarias y también, cuestiones socio-culturales.

#### **Mercantilización**

Los actuales mercados de consumo masivo provocan que los productos y servicios compitan de manera invisible al consumidor quien elige generalmente sólo en función del precio. Así, las organizaciones deben alcanzar mayores niveles de eficiencia e invertir en innovación para crecer. BPM aporta una buena herramienta para alcanzar estos objetivos.

#### **Productividad**

Las organizaciones deben producir más y mejores resultados en menos tiempo como una medida de su buena productividad. Sin lugar a dudas, la mejora continua como concepto inmerso en BPM es un buen mecanismo para autoevaluar a la organización y aumentar su productividad.

## Innovación

Si bien el término innovación se asoció durante mucho tiempo a la invención y al progreso, a nivel de negocio la innovación es mucho más ya que no sólo se aplica a productos sino también a servicios y modelos de negocio.

BPM es un concepto que sigue creciendo y sigue despertando el interés de investigadores. Los principales temas de investigación y desarrollo están relacionados a cómo dar respuesta a este concepto de innovación, incorporando características que vayan más allá del simple modelado de procesos industriales clásicos, cuyo resultado final es la obtención de un producto.

## Conformidad

El cumplimiento de normativas y regulaciones tiene un costo cada vez más creciente, siguiendo el ritmo de la globalización, la mercantilización y de la innovación. En este sentido la automatización de procesos que incorporen tal cumplimiento constituye un aporte fundamental para acompañar este crecimiento con reducción de costos.

## Cambios socio-culturales

En la era de la socialización del conocimiento, los trabajadores, las formas y las herramientas de trabajo se han transformado. La naturaleza del trabajo ha cambiado y conceptos como teletrabajo, colaboración, redes sociales y la creciente dependencia de la tecnología, ha permitido mayores niveles de productividad. Sin embargo, simultáneamente exigen nuevos enfoques.

El concepto de BPM social constituye una idea actual, abordada y analizada por esta tesis, como un crecimiento necesario de los procesos de negocio, su gestión y su ejecución.

## 2.2. DESARROLLO DIRIGIDO POR MODELOS

En esta sección se introducen conceptos vinculados al MDD, con el solo efecto de introducir al lector en un paradigma de construcción de software basado en un proceso guiado por modelos y donde la transformación entre los mismos constituye uno de los pilares. En MDD, son los metamodelos quienes definen la sintaxis abstracta de los lenguajes de modelado.

### 2.2.1. Orígenes de MDD y su Aplicación a BPM

El acrónimo MDD representa un paradigma de construcción de software basado en un proceso guiado por modelos y sus transformaciones. El adjetivo *driven* en MDD, a diferencia de *based*, enfatiza que este paradigma asigna a los modelos un rol central y activo: son al menos tan importantes como el código fuente. Los modelos se van generando, desde los más abstractos a los más concretos, a través pasos de transformación y/o refinamientos, hasta llegar al código aplicando una última transformación [39].

El MDD en el desarrollo de software tiene por objetivo separar la especificación funcional del software de su implementación en una plataforma ejecutable, a los efectos de mejorar la productividad, la portabilidad, la interoperabilidad y el mantenimiento y documentación. Entre las propuestas concretas para soportar MDD, se encuentran la arquitectura dirigida por modelos – (Model Driven Architecture, MDA) desarrollada por el Object Management Group (OMG) y los lenguajes de dominio específico (Domain-Specific Language, DSLs) impuestos por Microsoft. MDA tiende a ser más restrictivo, enfocándose en lenguajes de modelado basados en UML, mientras que los DSLs utilizan una notación propietaria para definir lenguajes.

En el contexto de BPM, la metodología integradora de servicios y procesos MISIP encuentra en el modelado de los procesos un punto de partida para explicitar el conocimiento y achicar la brecha entre los involucrados. Este trabajo propone formalizar cada etapa de dicha metodología a través de metamodelos y reglas de transformación entre dichos modelos.

### 2.2.2. Arquitectura Dirigida por Modelos

MDA es un concepto promovido por OMG desde noviembre de 2000, con el objetivo de afrontar los desafíos de integración de las aplicaciones y los continuos cambios tecnológicos. MDA es una arquitectura que proporciona un conjunto de guías para estructurar especificaciones expresadas como modelos, siguiendo el proceso MDD.

Dos de las principales características de MDA, es el concepto de plataforma y el de independencia de plataformas. Una plataforma en MDA es un conjunto de subsistemas y tecnologías que proveen un conjunto coherente de funcionalidades. Independencia de plataforma significa que un sistema debe ser modelado de modo que el modelo sea independiente de cualquier plataforma tecnológica. Los modelos abstractos son llamados modelos independientes de la plataforma (Platform Independent Model, PIM), en contraposición a modelos que incluyen detalles de la plataforma, llamados modelos específicos de la plataforma (Platform Specific Model, PSM).

El proceso de MDA involucra refinamientos sucesivos los cuales generalmente pueden clasificarse en:

- **Transformaciones** - son evoluciones que generan un nuevo tipo de modelo. Por ejemplo, transforman un PIM en un PSM o un PSM a código ejecutable. Cada transformación agrega detalles al modelo y también reducen el no-determinismo al tomarse decisiones de diseño. Por ejemplo, como representar datos, como implementar mensajes, etc.
- **Refinamientos** - son evoluciones que preservan la semántica del modelo, y producen el mismo tipo de modelo. Por ejemplo, refinar un PIM en un nuevo PIM.

Adicionalmente al desarrollo de software, los principios de MDA pueden aplicarse a otras áreas, como al modelado de procesos de negocio donde el PIM, independiente de la tecnología y de la arquitectura es adaptado tanto a los sistemas como a los procesos manuales.

### 2.2.3. Tipos de Modelos en MDD

Algunos modelos describen al sistema de manera independiente de los conceptos técnicos que involucra su implementación sobre una plataforma de software, mientras que otros modelos tienen como finalidad primaria describir tales conceptos técnicos. Teniendo en cuenta esta diferencia, los tipos de modelos que identifica MDD son:

- 1) **Modelo Independiente de la Computación (Computation Independent Model, CIM)** – Provee una vista del sistema desde un punto de vista independiente de la computación. Un CIM no muestra detalles de la estructura del sistema. Usualmente al CIM se lo llama modelo del dominio y en su construcción se utiliza un vocabulario que resulta familiar para los expertos de negocio en el dominio en cuestión. Se asume que los usuarios a quienes está destinado el CIM - los expertos de dominio - no poseen conocimientos técnicos acerca de los artefactos que se usarán para implementar el sistema. El CIM juega un papel muy importante en reducir la brecha entre los expertos en el dominio y sus requerimientos por un lado, y los expertos en diseñar y construir artefactos de software por el otro.
- 2) **Modelo Independiente de la Plataforma (Platform Independent Model, PIM)** – Es un modelo con un alto nivel de abstracción que es independiente de cualquier tecnología o lenguaje de implementación. Dentro del PIM el sistema se modela desde el punto de vista de cómo se soporta mejor al negocio, sin tener en cuenta como va a ser implementado: ignora los sistemas operativos, los lenguajes de programación, el hardware, la topología de red, etc. Por lo tanto un PIM puede luego ser implementado sobre diferentes plataformas tecnológicas.
- 3) **Modelo Específico de la Plataforma (Platform Specific Model, PSM)** – El próximo paso de evolución de un PIM, es uno o más PSM (PSM - Platform Specific Models). Un PIM representa la proyección de los PIMs en una plataforma específica. Un PIM puede generar múltiples PSMs, cada uno para una tecnología en particular. Generalmente, los PSMs deben colaborar entre sí para una solución completa y consistente. Por ejemplo, un PSM para JAVA contiene términos como clase, interfase, etc. Un PSM para una base de datos relacional contiene términos como tabla, columna y clave foránea.
- 4) **Modelo de Implementación (Implementation Model, Código)** - El paso final en el desarrollo es la transformación de cada PSM a código fuente. Ya que el PSM está orientado al dominio tecnológico específico, esta transformación es bastante directa.

## 2.2.4. La Arquitectura de Cuatro Capas

El metamodelado es un mecanismo que permite definir formalmente lenguajes de modelado, como por ejemplo, UML y el Object Constraint Language (OCL). La Arquitectura Cuatro Capas de modelado es la propuesta de la OMG orientada a estandarizar conceptos relacionados al modelado, desde los conceptos más abstractos a los más concretos, o cercanos a la implementación.

Los cuatro niveles definidos en esta arquitectura se denominan: M0 (más concreto), M1, M2, M3 (más abstracto).

### Nivel M0 – Instancias

En el nivel M0 se encuentran todas las instancias “reales” del sistema, es decir, los objetos de la aplicación. Aquí no se habla de clases, ni atributos, sino de entidades físicas que existen en el sistema.

### Nivel M1 – Modelo del Sistema

Por encima de la capa M0 se sitúa la capa M1, que representa el modelo de un sistema de software. Los conceptos del nivel M1 representan categorías de las instancias de M0. Es decir, cada elemento de M0 es una instancia de un elemento de M1.

### Nivel M2 – Metamodelo

Análogamente a como ocurre con las capas M0 y M1, los elementos del nivel M1 son a su vez instancias del nivel M2.

### Nivel M3 – Meta-Metamodelo

De la misma manera podemos ver los elementos de M2 como instancias de otra capa, la capa M3 o capa de meta-metamodelo. Un meta-metamodelo (OMG, 2003) es un modelo que define el lenguaje para representar un metamodelo. La relación entre un meta-metamodelo y un metamodelo es análoga a la relación entre un metamodelo y modelo. Es el nivel más abstracto, que permite definir metamodelos concretos.

Dentro de los estándares de la OMG, el Meta-Object Facility (MOF) es el lenguaje estándar de la capa M3. Esto supone que todos los metamodelos de la capa M2, son instancias de MOF. Es claro que UML no es el único lenguaje de modelado, sino que otros lenguajes específicos de dominio y lenguajes de modelado estandarizados pueden ser definidos sobre la base de MOF. MOF puede ser usado para definir metamodelos orientados a objetos, como es el caso de UML, y también para otros metamodelos no orientados a objetos, como es el caso de las redes de Petri o metamodelos para servicios web.

**Definición 4. LENGUAJE META-OBJECT FACILITY o MOF**

*El lenguaje MOF es un estándar de la OMG para la ingeniería conducida por modelos.*

Tal como su nombre lo indica, MOF se basa en el paradigma de Orientación a Objetos. Por este motivo usa los mismos conceptos y la misma sintaxis concreta que el core de UML.

**2.2.5. Uso del Metamodelado en MDD**

El metamodelado es un mecanismo para definir lenguajes de modelado sin ambigüedades y permitir que una herramienta de transformación pueda leer, escribir y entender modelos. En particular, los metamodelos pueden transformarse a través de reglas de transformación. Las mismas constituyen una definición de la transformación y describen, como un modelo de un lenguaje fuente puede ser transformado a un modelo en el lenguaje destino. Además, los lenguajes en los cuales se expresan las reglas de transformación también deben estar formalmente definidos para permitir su automatización.

Como se vió anteriormente, MOF se encuentra en la capa superior de la arquitectura de 4 capas. Provee un meta-meta lenguaje en la capa superior que permite definir metamodelos en la capa M2. El ejemplo más popular de un lenguaje en la capa M2 es el metamodelo UML, que describe el lenguaje UML.

MOF es una arquitectura de metamodelado cerrada. Esto significa que el metamodelo MOF se define en términos de si mismo. Asimismo, MOF permite una arquitectura de metamodelado estricta, es decir, cada elemento de un modelo en cualquiera de las capas tiene una correspondencia estricta con un elemento del modelo de la capa superior.

**2.3. COMPUTACIÓN EN LA NUBE**

En esta sección se detallan las bases conceptuales de la computación en la nube (*cloud computing*), revisando primero los antecedentes tecnológicos que originaron este nuevo modelo de computación. Se describen luego características esenciales, los diferentes modelos de servicio y de despliegue. Finalmente se enuncian algunas ventajas y desventajas de la computación en la nube y su relación con BPM.

**2.3.1. Definiciones**

Una de las definiciones más referenciadas es la provista *por el* National Institute of Standards and Technology (*NIST*) [17]:

**Definición 5. COMPUTACION EN LA NUBE [NIST]**

*La computación en la nube es un modelo que habilita, de manera ubicua y bajo demanda, el acceso a la red para compartir un conjunto de recursos de computación configurable (redes, servidores, almacenamiento, aplicaciones y servicios) que pueden ser provistos y publicados con un mínimo esfuerzo de administración por parte del proveedor del servicio o de interacción con el mismo.*

Otra definición provista por la European Network and Information Security Agency (ENISA), enuncia [18]:

**Definición 6. COMPUTACION EN LA NUBE [ENISA]**

*La computación en nube es un modelo de servicio bajo demanda para la prestación de TI, a menudo basado en la virtualización y en las tecnologías informáticas distribuidas. Las arquitecturas de computación en nube poseen: 1) recursos con un alto grado de abstracción; 2) escalabilidad y flexibilidad prácticamente instantáneas; 3) prestación casi instantánea; 4) recursos compartidos (hardware, base de datos, memoria, etc.); 5) servicio bajo demanda que suele incluir un sistema de facturación de pago por uso; y 6) gestión programática, como por ejemplo, mediante la API del servicio web.*

Si bien las definiciones anteriores son consideradas como las más completas y conocidas, en [26] un grupo de reconocidos expertos, presentan más definiciones. Algunas de ellas, se presentan a continuación. Por ejemplo, la definición provista por Furht, Borko, Escalante, y Armando, en su libro “Handbook of Cloud Computing” es la siguiente:

**Definición 7. COMPUTACION EN LA NUBE [FURHT ET AL.]**

*Computacion en la Nube puede definirse como un nuevo estilo de computación en el cual recursos dinámicamente escalables y usualmente virtualizados son provistos como servicios a través de Internet.*

La definición de UC Berkeley RAD Labs indica lo siguiente:

**Definición 8. COMPUTACION EN LA NUBE [UC Berkeley RAD Labs.]**

*Computacion en la Nube es un nuevo enfoque de computación caracterizado por: 1) la ilusión de contar con infinitos recursos de cómputos, 2) la eliminación de compromisos de antemano por parte de los usuarios de la Nube, y 3) la habilidad de pagar por el uso, como sea necesario.*

Por otra parte, McKinsey & Company enuncia:

### Definición 9. COMPUTACION EN LA NUBE [McKinsey]

*La Nube son servicios basados en hardware que ofrecen capacidades de cómputos, de redes y de almacenamiento donde la administración del hardware es altamente abstraída de los compradores, los compradores incurren en costos de infraestructura como una variable OPEX (costos operativos), y la capacidad de infraestructura es altamente flexible.*

La comparación de las definiciones anteriores se presenta en la Tabla 2. Los aspectos comunes incluyen un nuevo modelo de computación, la posibilidad de acceder a recursos ubícuos, escalables y configurables por Internet, la abstracción del hardware y el pago por uso de recursos computacionales.

**Tabla 2. Comparación de Definiciones de Computación en la Nube**

DEFINICIÓN	CARACTERÍSTICAS RELEVANTES	ASPECTOS COMUNES
NIST	<ul style="list-style-type: none"> <li>○ Es un modelo</li> <li>○ Acceso a recursos configurables</li> <li>○ Acceso a recursos ubícuos</li> <li>○ Acceso a recursos compartidos</li> <li>○ Mínimo esfuerzo de administración</li> <li>○ Mínimo esfuerzo de interacción</li> </ul>	<ul style="list-style-type: none"> <li>○ Nuevo modelo de Computación</li> <li>○ Acceso a recursos ubícuos</li> <li>○ Recursos escalables</li> <li>○ Recursos configurables</li> <li>○ Acceso a recursos por Internet</li> <li>○ Abstracción del hardware</li> <li>○ Pago por uso</li> </ul>
ENISA	<ul style="list-style-type: none"> <li>○ Modelo de servicio bajo demanda</li> <li>○ Recursos con alto grado de abstracción</li> <li>○ Recursos escalables</li> <li>○ Recursos flexibles</li> <li>○ Prestación inmediata</li> <li>○ Recursos compartidos</li> <li>○ Servicio bajo demanda</li> <li>○ Pago de servicios por uso</li> <li>○ Gestión para programar los servicios</li> </ul>	
Furht et al.	<ul style="list-style-type: none"> <li>○ Nuevo estilo de computación</li> <li>○ Recursos escalables</li> <li>○ Recursos virtualizados</li> <li>○ Recursos como servicios</li> <li>○ Acceso a recursos por Internet</li> </ul>	
UCBerkeley	<ul style="list-style-type: none"> <li>○ Nuevo enfoque computacional</li> <li>○ Idea de recursos infinitos</li> <li>○ Sin compromisos previos</li> <li>○ Pago por uso</li> </ul>	



### 2.3.2. Antecedentes Tecnológicos

La computación en la nube es un paradigma que posibilita el acceso ubicuo bajo demanda a servicios IT accesibles a través de Internet. El término nube (*cloud*) se refiere a la forma de representar la red (Internet) en los diagramas y es una abstracción de las complejidades de su infraestructura [16].

El desafío es cómo dimensionar la capacidad de los servidores para ajustarlos a la demanda (servicios estacionales, eventos masivos y puntuales, juegos on line). La Figura 3 representa alguno de los elementos involucrados dentro de “la nube” y que deben funcionar como un sistema único y de manera transparente. La misma muestra distintos usuarios, canales de acceso y dispositivos, y una extensa variedad de servicios existentes.



**Figura 3. Elementos de Computación en la Nube**

Una arquitectura de tal complejidad, provoca incertidumbre acerca de la demanda real de recursos exigida por los usuarios y las aplicaciones. Esto

puede provocar una mala utilización de los recursos, donde puede haber picos de demanda insatisfecha en ciertos momentos y recursos ociosos en otros.

El paradigma de la nube se basa en la noción de aprovechamiento bajo demanda. Esto significa, consumir lo que se necesite y pagar por lo que se consume.

Los primeros proveedores de servicios en la nube surgen de la evolución de la infraestructura privada de los propios proveedores de servicios de Internet, o Internet Service Providers (ISPs) y lo hicieron como una manera de aprovechar recursos ociosos en sus datacenters, abriendo las interfaces de gestión a los usuarios y agregando lo necesario para implementar un modelo de pago por uso. Estos proveedores incluyen a Amazon, Google, y Microsoft, entre otros.

Las infraestructuras de computación en la nube se encuentran soportadas por servidores tradicionales y virtualizados. En este sentido se puede afirmar que la virtualización es uno de los conceptos tecnológicos que sustentan la infraestructura. A este concepto de virtualización se suman otras tecnologías relacionadas como por ejemplo, Service Oriented Architecture (SOA), Grid Computing y Servicios Web. Asimismo, cabe destacar que la nube no presenta una nueva tecnología, sino un nuevo paradigma de uso que combina tecnologías existentes. En síntesis, la nube no incorpora aspectos tecnológicos nuevos sino que ofrece un nuevo modelo de servicio combinando lo existente.

#### 2.3.4. El Modelo de Nube

El modelo de nube enunciado por NIST responde a la metáfora 5-3-4 que definen 5 características esenciales, 3 modelos de servicio y 4 modelos de despliegue.

Las cinco características esenciales incluyen:

- 1) **Autoservicio a demanda** – un consumidor puede aprovisionar unilateral y automáticamente recursos de computación según necesite.
- 2) **Amplio acceso a la red** – las capacidades están accesibles en la red a través de mecanismos estándar que permiten el acceso desde plataformas heterogéneas de clientes.
- 3) **Recursos mancomunados** – los recursos de computación provistos son agrupados para servir a múltiples clientes usando un modelo multi-inquilino
- 4) **Elasticidad rápida** – las capacidades son aprovisionadas y liberadas rápida y elásticamente, y en algunos casos automáticamente.
- 5) **Servicio medido** – los sistemas en la nube controlan y optimizan automáticamente el uso de los recursos proporcionando alguna capacidad de medición, habitualmente basado en pago por uso o cargo por uso.

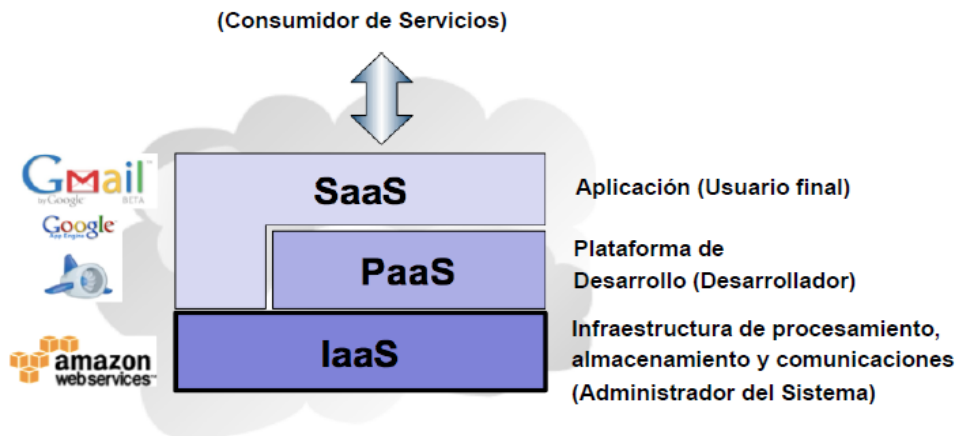
Los tres modelos de servicios incluyen:

- 1) **Infraestructura como Servicio** – en inglés, Infrastructure as a Service (IaaS) donde el consumidor aprovisiona recursos de computación (e.g. capacidad de CPU, almacenamiento, red) en los que ejecuta su software, incluidas aplicaciones y sistemas operativos. El consumidor no controla la infraestructura de nube subyacente pero sí los sistemas operativos, el almacenamiento, las aplicaciones desplegadas y a veces la red (firewalls).
- 2) **Plataforma como Servicio** – en inglés, Platform as a Service (PaaS) donde el consumidor despliega en la infraestructura provista por el proveedor, aplicaciones tanto propias como adquiridas, desarrolladas usando entornos de programación soportados por el proveedor.
- 3) **Software como Servicio** – en inglés, Software as a Service (SaaS) donde el consumidor utiliza las aplicaciones del proveedor que son ejecutadas en una infraestructura de nube. El consumidor no controla ni la infraestructura de nube subyacente, ni las capacidades de la aplicación, pero puede controlar la configuración personal de dichas aplicaciones.

Los cuatro modelos de despliegue incluyen:

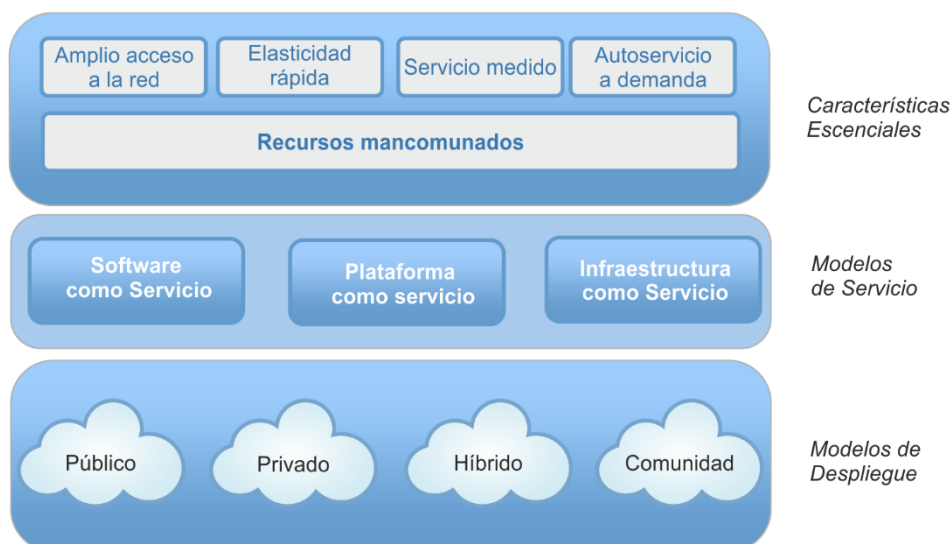
- 1) **Nube Pública** – Ofrece una infraestructura para uso del público o empresas en general. Estas nubes tienen una interface para la gestión interna de la infraestructura virtualizada y exponen otra externa para la gestión de los recursos de los clientes.
- 2) **Nube Privada** – Es para uso exclusivo de una organización con múltiples clientes (e.g. departamentos de una empresa). Puede ser de su propiedad (on-premise), alquilada (off-premise) o una combinación de ambas. Permite una gestión flexible y ágil de los recursos de la organización. Los principales proveedores públicos de IaaS comenzaron creando nubes privadas para mejorar la gestión de sus datacenters.
- 3) **Nube Híbrida** – Es la combinación de nube privada y pública que permite gestionar los picos de carga obteniendo recursos de la nube pública. El uso de una nube pública es totalmente transparente para usuarios de la nube privada.
- 4) **Nube Comunitaria** - Es una infraestructura para uso de una comunidad de organizaciones que comparten intereses, donde cada componente conserva su autonomía. La relación entre componentes se hace mediante tecnologías (propietarias o estandarizadas) que permitan la portabilidad de datos y aplicaciones. Puede ser propiedad de una o varias de las organizaciones, ser alquilada o una combinación de ambas.

La Figura 4 resume el modelo de nube descrito ubicando cada uno de los modelos que lo componen de manera integrada.



**Figura 4. Modelo de Nube. Visión Integrada**

La Figura 5 muestra el modelo de nube provisto por NIST de manera integrada incluyendo también las 5 características esenciales.



**Figura 5. Modelo de Nube [NIST]**

### 2.3.5. Análisis del Modelo de Nube y su Aplicación en BPM

El modelo de nube presenta ventajas y desventajas en general, que a la hora de ser analizadas a la luz de BPM deben ser consideradas a fin de proponer nuevos modelos de servicio denominados Business Process as a Service (BPaaS) [28]. A continuación se analiza el modelo de nube y su aplicación en BPM.

En general, el modelo de nube presenta las siguientes ventajas:

- Reducción de costos – el modelo de pago por uso resulta beneficioso en cuanto la inversión requerida para montar una infraestructura tecnológica, sobre todo en el ámbito de pequeñas y medianas empresas que puján en un

mercado competitivo y donde la innovación es un aspecto esencial para su crecimiento, tal como se detalló en la sección 2.1.5.

- Mejora de la eficiencia – es posible dedicar más recursos a la actividad propia de una organización delegando la responsabilidad de la gestión de TI, además de facilitar la escalabilidad y la elasticidad en el aprovisionamiento de recursos tecnológicos.
- Flexibilidad – se evita la inversión en hardware que se desactualiza rápidamente, además de permitir las operaciones desde cualquier lugar.
- Seguridad – los proveedores de servicios en la nube generalmente poseen mejores prácticas y sistemas de seguridad, así como aseguran el cumplimiento de regulaciones.
- Continuidad del negocio – la alta fiabilidad y la tolerancia a fallos, provista por sistemas redundantes, replicación de datos y distribución geográfica permiten a las organizaciones otorgar un servicio 24x7.

Asimismo, el modelo de nube presenta las siguientes desventajas:

- Seguridad – así como en un aspecto, la seguridad es una ventaja en el modelo de nube, se transforma en desventaja cuando se observa la seguridad física de los datacenters, las garantías de transmisión cifrada y el acceso a datos sensibles para la organización.
- Dependencia de la red – la necesidad de estar siempre conectado para poder trabajar, puede ser una desventaja, así como la dependencia de las velocidades de la red.
- Dependencia del vendedor – la fiabilidad del proveedor, así como los fallos masivos y la escasa interoperabilidad por la carencia de estándares, constituyen sin dudas un riesgo en la operación de las compañías.
- Aspectos legales – el modelo de nube aún carece de adecuada armonía entre legislaciones de distintos países, más aún cuando se piensa que la muchas de estas arquitecturas operan con modelos globalizados. Asimismo, es poco clara aún la protección al consumidor.
- Integridad de datos – la pérdida de control de los datos de las organizaciones pone en alto riesgo la privacidad y confidencialidad de los mismos.

Teniendo en cuenta las ventajas y desventajas descriptas anteriormente, a continuación se discute la aplicación de este modelo a BPM.

BPM basada en la nube da a los usuarios la posibilidad de usar software de una manera de “pago por uso”, en lugar de forzarlos a emprender grandes

inversiones en software de BPM, hardware y mantenimiento como se da en el caso de los sistemas de licenciamiento tradicional. Por otra parte, los sistemas pueden escalarse, creciendo o decreciendo, de acuerdo a las necesidades de los usuarios, lo cual significa que los mismos no deben preocuparse acerca del sobre o bajo aprovisionamiento de recursos, gracias a la alta adaptabilidad provista en la actualidad por los prestadores de servicios en la nube.

Sin embargo, existentes puntos débiles: al desplegar un BPMS en la nube, los usuarios pueden llegar a perder control sobre los datos sensibles. Este aspecto resulta no menor al considerar que los procesos de negocio dentro de una organización gestionan información de gran importancia para la misma y sus miembros. Por otro lado, la eficiencia y efectividad de las actividades de los procesos que no son altamente computacionales pueden no incrementarse por ponerlas en la nube, sino al contrario, estas actividades pueden volverse más costosas. Por ejemplo, una actividad que no es intensamente computacional podría necesitar procesar cierta cantidad de datos. La transferencia de los datos a la nube puede tomar más tiempo que transmitirlos a una versión embebida instalada localmente (on-premise). Incluso la misma puede resultar aún mayor que la necesidad real de procesamiento. Además, los costos de la actividad pueden incrementarse debido a que la transferencia de datos es uno de los elementos de facturación en un sistema de computación en la nube, debido a la alta disponibilidad de la conexión [145].

## 2.4. SOFTWARE SOCIAL

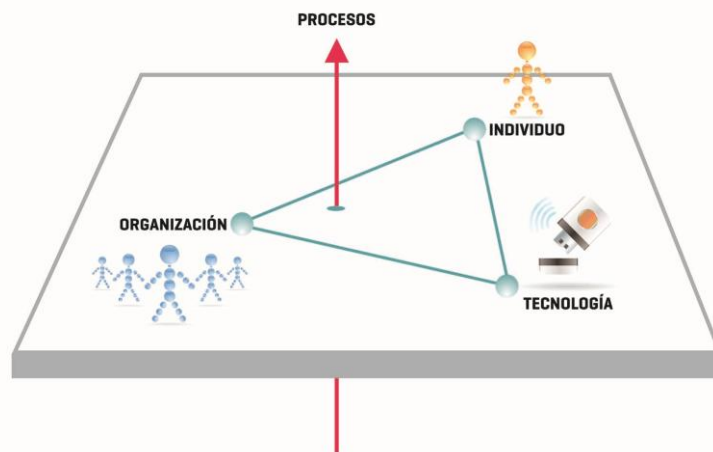
En esta sección se presentan conceptos de software social y su evolución, con la mirada dirigida a las organizaciones que ya han comenzado a evaluar la importancia de considerar estas ideas como parte de su funcionamiento operativo y también como mecanismos que permitan absorber mejor y más rápidamente los cambios del entorno.

El origen del concepto de software social data de 1987. En ese año, en [19], se plantea la existencia del concepto enunciado por Eric Drexler [21], pero cobraría mayor relevancia en 2002 cuando Clay Shirky lo define como "el software que soporta la interacción grupal". Expandiendo esta idea, Adina Levin dice que se trata de "el conjunto de herramientas para facilitar la interacción y la colaboración, que dependen más de las convenciones sociales (en su uso) que de las propias funcionalidades que ofrecen". Más recientemente, en el 2003, en [22] se afirma que el software social es software construido a partir de una o más de las siguientes premisas: 1) dar soporte a la interacción conversacional entre individuos o grupos –incluyendo conversaciones en tiempo real o diferido, e.g. mensajería instantánea y espacios de colaboración para equipos de trabajo, respectivamente; 2) dar soporte a la realimentación social –que permita a un grupo valorar las contribuciones de otros, quizás implícitamente, permitiendo la creación de una reputación digital; 3) dar soporte a las redes sociales –para crear y gestionar explícitamente una expresión digital de las relaciones personales de los individuos, así como para ayudarlos a crear nuevas relaciones.

Utilizar software social como estrategia para gestionar el conocimiento global ha resultado de interés tanto a niveles organizacionales como educativos. Sin embargo, en lugar de ser proactivo, las organizaciones tienden a enfrentar las barreras relacionadas con la gestión del conocimiento después de que ocurren los problemas. Cuando se trata de las tecnologías sociales en un entorno distribuido, las organizaciones y los individuos se enfrentan a una serie de barreras que son consecuencia de la incapacidad de afrontar la complejidad.

#### 2.4.1. Las Organizaciones y el Software Social

Las organizaciones son vistas como un sistema. Tomando las premisas de la Teoría General de Sistemas (TGS), se puede entender la complejidad de las mismas como la cantidad de elementos que componen la organización (complejidad cuantitativa) y como las potenciales interacciones y estados posibles que se producen desde las mismas. En [19] se propone un posible modelo de complejidad organizacional denominado Organización, Individuo, Tecnologías y Procesos (OITP). El acrónimo OITP se refiere a los cuatro elementos (o dimensiones) a considerar en una situación de complejidad: 1) la Organización (O) - considerando la estructura funcional y orgánica de la empresa, estructuras de poder, circuitos de decisión, etc.; 2) los Individuos (I) – incluyendo al resto de personas no consideradas en la Organización; 3) la Tecnología (T) - comprende los propios objetos tecnológicos, los métodos, técnicas y a los propios técnicos; y 4) los Procesos (P) de negocio, que representan la actividad de la empresa. El modelo OITP se muestra en la Figura 6. La relevancia de este modelo para este trabajo radica en que incluye a los procesos como uno de los elementos que componen la organización, y lo hace con una visión transversal, tal como lo plantea el paradigma BPM.



**Figura 6. Modelo OITP**

La Figura 7 completa el modelo OITP teniendo en cuenta el aspecto tecno-social, que se ve reflejado en el software social.

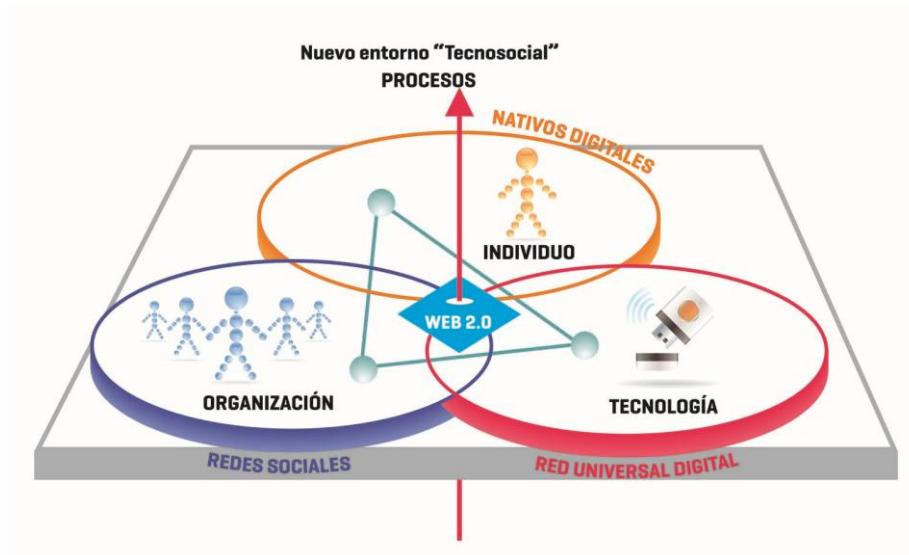


Figura 7. Modelo OITP con Aspecto Tecno-Social

#### 2.4.2. Web 2.0 y el Software Social

La Web 2.0 puede entenderse como una forma innovadora de introducir inteligencia a la Web y que se logra involucrando directamente al usuario como parte del sistema. Es un fenómeno que ha irrumpido con fuerza en los medios de comunicación y en la manera de comunicarse. En [19] se afirma que se trata de *“el proceso de emergencia de una Web de Nueva Generación (WebNG) que se integra en un proceso de mayor alcance como es la aparición de un Nuevo Entorno Tecnosocial sustentado sobre una Red Universal de la que Internet y la propia Web son sólo la parte más visible”*.

Las características más importantes provenientes de la Web 2.0 y que han impactado en lo que denominamos Software Social son [19]:

- **Interactividad** – La interacción entre los componentes del sistema, considerando ya al usuario como parte del mismo, trasciende las características habituales de pregunta-respuesta para convertirse en un esquema conversacional que empuja a la socialización y que tiende a que pase a formar parte del ecosistema natural entre individuos y por ende, en las organizaciones.
- **Rastreabilidad** – El incremento de la interacción en la red, tiende a dejar huellas que deben poder ser recuperadas, administradas y utilizadas como parte de la información que alimenta a los sistemas. Como ejemplos de interacciones existentes podemos mencionar la operación de una etiqueta activa, por ejemplo con tecnología Radio Frequency IDentification (RFID), las transacciones financieras realizadas a través de Internet, un comentario realizado en un blog o en un foro, o una etiqueta en una red social.



- **Virtualidad** – Este concepto nos lleva a comprender que más que interactuar con una realidad virtual, se vive en una virtualidad real, donde, el ejemplo más evidente es el dinero que fluye por circuitos financieros globales de manera virtual.
- **Ubicuidad** – Este concepto se refiere al acceso no localizado, desde cualquier ubicación geográfica que puede acortar brechas sociales y culturales y que impactan en la manera de concebir a las organizaciones.
- **Movilidad** – Esta característica, muy relacionada con la ubicuidad, ha derivado en nuevos usos y costumbres de los usuarios, vistos tanto como integrantes de una organización o como consumidores.

### 2.4.3. Evolución del Software Social

Según el enfoque planteado para presentar el concepto de software social y en vistas a considerar su vinculación con el paradigma BPM, resulta de interés plantear brevemente la evolución de este concepto. Para ello, a continuación se describen distintos tipos de software social.

#### Blogs

Los Blogs como herramientas personales de comunicación y expresión individual en la Red, son un fenómeno social que acercó la Web a los individuos y socializó la misma. Según se enuncia en [19] “la definición de blog (weblog o bitácora), va desde el simple diario personal en Internet, hasta una herramienta de expresión, comunicación y socialización, que ha servido para tejer un complejo subespacio de comunicación en la Red que conocemos como blogosfera”. Ejemplos de Blogs pueden ser: Blogger, BlogSpot

#### Wikis

Las wikis se han inspirado en el concepto original de Web como una trama hiperenlazada de alcance global y con formato de lecto-escritura y surgen como un mecanismo de documentación para la ingeniería de software donde el versionado de los documentos y la trazabilidad fiable de la historia de los documentos resulta una de las características más importantes.

Este concepto alcanza su máxima expresión con el lanzamiento en 2001 de Wikipedia una enciclopedia de elaboración cooperativa y distribuida y de acceso gratuito. Este proyecto específico ha llegado a desarrollarse hasta el punto de convertirse en el foco de agrios debates acerca de la generación y creación colectiva de conocimiento, llegando a confundirse, tanto en alcance como definición, con el término genérico de wiki. Ejemplos de Wikis: Wikipedia es la Wiki más popular o Media-Wiki.

#### Redes Sociales

El concepto de red social es una evolución natural del concepto de blog donde el resultado es un sitio con contenido generado por el usuario o User Generation

Content (USG) [23], donde se permite la participación a escala masiva. Los individuos se conectan unos a otros a través de una red conformada por intereses compartidos, objetivos profesionales comunes o conocidos mutuos. Cuando un usuario se une a una red social conforma un espacio propio en la red, pero con características dinámicas y colaborativas y que además expone a los individuos hacia el mundo exterior. Este esquema permite escuchar, conversar y también influir en los comportamientos y acciones. Este enfoque si bien es individual, su masificación obliga a ser considerado a nivel de las organizaciones que replantean su funcionamiento y evalúan su adopción bajo las siguientes acciones: 1) Identificar los usuarios claves en los grupos; 2) conocer a los usuarios claves, sus dinámicas, funcionamiento y motivaciones; 3) convertir a los usuarios claves en evangelistas y a estos en formadores; y 4) apoyar la adopción de abajo-arriba y las iniciativas emergentes. Ejemplos de redes sociales incluyen: Facebook, Twitter, LinkedIn, Google+, Xing

## **2.5. BPM Y COMPUTACIÓN EN LA NUBE**

En las secciones previas hemos analizado las características y conceptos básicos de la metodología BPM y del modelo de computación en la nube, como dos temáticas actuales y que son el foco de atención de las organizaciones que deben poner atención en mejorar su negocio a la luz de la globalización, la innovación, y la rápida absorción de los cambios, dentro de un mercado cada vez más competitivo y emergente.

En esta sección se analiza la convergencia o divergencia entre estos dos conceptos y se establecen las bases para definir criterios que den sentido a la utilización de la computación en la nube dentro de la metodología BPM.

### **2.5.1. BPM en el Nube - Implicancias y Consideraciones**

La utilización de BPM en la nube ha producido un cambio en la concepción tanto del diseño como de la ejecución de los procesos de negocio. Los entornos distribuidos en este contexto ofrecen posibilidades de cómputo que resultan ventajosas para los procesos, y las mismas resultan mayormente aprovechables al ser consideradas bajo un esquema de descomposición. Este concepto ha sido introducido en BPM permitiendo a los procesos de negocio ejecutarse tanto en un ambiente de nube como en uno embebido (on-premise), aprovechando así las ventajas de ambos esquemas bajo la consideración de criterios relevantes, como por ejemplo, la sensibilidad en los datos, altas prestaciones de cómputo y la portabilidad de los sistemas intervinientes [145].

Si bien BPM constituye una metodología ampliamente adoptada por las organizaciones, cuando se analiza dentro de un modelo de computación, se deben considerar las tecnologías y herramientas que facilitan su adopción. Por ejemplo, la adopción de un BPMS en una organización puede constituir un proyecto costoso tanto desde el punto de vista de la inversión que debe realizarse en software y en hardware, así como en los recursos humanos entrenados para desplegar y mantener la solución. Por otra parte, la

escalabilidad puede ser un aspecto a considerar en estas organizaciones dado que el motor de procesos es capaz de atender simultáneamente una cantidad limitada de instancias de procesos [24].

En este sentido, la necesidad de escalar y la naturaleza netamente colaborativa que subyace a la ejecución de procesos de negocio, ponen de manifiesto la necesidad de contar con entornos descentralizados que permitan optimizar el uso de la tecnología y dar respuesta a la necesidad de crecimiento de las organizaciones. Es así que las arquitecturas basadas en la nube ofrecen una buena alternativa por su enfoque flexible y ágil. En particular, los procesos de negocio colaborativos en los cuales interactúan distintos sectores de una misma o de diferentes organizaciones, ofrecen un potencial de optimización a través de la combinación de técnicas de computación en la nube y BPM.

A la luz del ciclo de vida de los procesos de negocio presentados en la sección 2.1.2 y analizadas las características del modelo de computación en la nube, se puede decir que hay dos puntos sensibles dentro de este ciclo de vida que están en contacto directo con la nube: el diseño, y la ejecución y monitoreo. Las características para aplicar computación en la nube durante ambas etapas se explican a continuación.

### **Diseño de Procesos en la Nube**

Como se ha presentado en la Sección 2.1.3, el análisis y diseño de procesos de negocio produce como resultado un modelo de proceso de negocio. Estos modelos constituyen la piedra fundamental para instrumentar los procesos de negocio en una organización. Las notaciones gráficas facilitan la representación de estos modelos.

La construcción de un modelo de negocio no es una actividad aislada, sino por el contrario, constituye la etapa del ciclo de vida de los procesos donde más interacción existe entre los involucrados (stakeholders), es por lo tanto una tarea colaborativa.

La noción de trabajo colaborativo no es nuevo en dentro en el área de TI y ha sido motivo de estudio durante muchos años. Esta idea de colaboración ha adquirido nuevo sentido con el advenimiento de la computación en la nube, no solamente por su transparencia de la ubicación física sino por su capacidad de gestionar versiones de cambios y verificar su consistencia.

En este sentido, el uso de la computación en las fases de análisis y diseño conducen a la necesidad de contar con herramientas de modelado colaborativas y en la nube bajo el modelo SaaS, donde su utilización permite que todos los involucrados puedan visualizar, modificar y consultar los modelos con el sólo uso de un navegador.

### **Ejecución de Procesos en la Nube**

La ejecución de procesos se vale de una arquitectura tecnológica donde el protagonista es el motor de procesos que almacena la definición de los procesos

de negocio, lanza sus instancias y monitorea su ejecución. La salida de esta ejecución monitoreada constituye el insumo principal para cerrar el ciclo de mejora continua.

Con la adopción de la computación en la nube en el marco de BPM surge la necesidad de dividir los procesos de negocio en procesos colaborativos individuales que puedan ejecutarse indistintamente de manera embebida o en la nube. Esto es lo que se conoce como descomposición de procesos, siendo esta un área extensamente analizada en la literatura actual y que se detalla en la sección 2.5.4.

Sin embargo, esta descentralización desafía a la arquitectura centralizada tradicional de los BPMS cuya componente fundamental, los motores de procesos, constituyen una componente centralizada, fundamentalmente en cuanto a su capacidad de control y monitoreo en un entorno distribuido.

### **2.5.2. Condiciones para Utilizar BPM en la Nube**

Con el rápido desarrollo de TI en el contexto del lanzamiento y ejecución de arquitecturas basadas en la nube, las compañías se enfrentan con nuevos problemas. En particular, los procesos de negocio colaborativos ofrecen un potencial de optimización a través de la combinación de técnicas de computación en la nube y BPM. Un factor común entre ambos conceptos es el enfoque flexible y ágil. El paradigma de computación en la nube puede considerarse un facilitador de la combinación mejorada de arquitecturas orientadas a servicios, así como un procedimiento ágil con respecto a la gestión de procesos de negocio. Pero este potencial depende de las condiciones de los distintos frameworks, las cuales pueden ser apreciadas desde un aspecto técnico tanto como económico. A continuación, se discuten estos dos aspectos.

#### **Vista Técnica**

Desde un punto de vista técnico se pueden identificar tres dimensiones para el diseño, implementación y operación exitosos de herramientas de BPM en un ambiente de nube: 1) programación, 2) integración y 3) seguridad.

- 1) **Programación** – Los sistemas complejos y distribuidos son altamente realizables en el campo de TI actual. En conexión con el objetivo de alcanzar una mayor usabilidad y flexibilidad, esta complejidad representa nuevos requerimientos para la Ingeniería de Software. Para resolver este problema es necesaria la adopción de nuevos lenguajes, recayendo sobre nuevos conceptos y técnicas innovadoras.
  
- 2) **Integración** – La integración puede dividirse en integración de datos, integración de funciones e integración de procesos. A la luz de los desafíos involucrados, el tópico de la integración juega un rol fundamental en distintos escenarios. Por ejemplo, un workflow basado en la nube puede controlar actividades variables distribuidas más allá de las fronteras de las compañías.

Para una ejecución no problemática de varias instancias de proceso se necesitan interfaces de integración y métodos estructurados.

- 3) **Seguridad** – La seguridad puede dividirse en tres categorías: seguridad funcional, de la información y de los datos. Todas estas categorías tienen una relevancia significativa para BPM, especialmente en cuanto a los *grids* de servidores de procesos de negocio. La seguridad funcional específica como el estado actual se corresponde con el estado deseado de funcionalidad. La seguridad de información se enfoca en los cambios o extracciones de información no autorizados. La seguridad de los datos se encarga de los datos relacionados con el proceso.

Bajo estos criterios, desde un punto de vista técnico, se debe investigar qué procesos de negocio son más apropiados para ejecutar en una arquitectura basada en la nube. Posibles inconvenientes a este despliegue son la existencia de opciones insuficientes de integración, o interfaces de programación de aplicaciones que deban ser tomados en consideración [25] [26] [27].

### Vista Económica

Se pueden mencionar dos dimensiones desde el punto de vista económico:

- 1) **Disponibilidad** – Los servicios que son provistos por una infraestructura de nube pueden ser accedidos en cualquier momento. Basados en un alto nivel de abstracción, la personalización e instalación se vuelven significativamente más fáciles. En adición a esta simplificación, el usuario final es capaz de trabajar con el servicio en forma inmediata.
- 2) **Riesgo de inversión** – En el contexto de los distintos modelos de facturación variable como el de pago por transacción, el sistema orientado a la nube resulta más accesible que un sistema de licenciamiento tradicional.

### 2.5.3. Combinación Nube y Embebido

La protección de la privacidad es una de las barreras para ejecutar BPM en un ambiente de nube. No todos los usuarios desean poner sus datos sensibles en la nube. Además es necesario observar la portabilidad de productos y versiones, y su disponibilidad en un esquema de nube.

Otro aspecto a considerar es la eficiencia. Las actividades de cómputo intensivo se pueden beneficiar en la nube debido a la escalabilidad y a la alta disponibilidad de fuerza de cómputo. Las tareas que no sean de cómputo intensivo, por otra parte, no siempre se benefician en este entorno. La performance de una actividad que está corriendo en un entorno embebido debería ser mayor que en la nube debido a la cantidad de datos que deben ser transferidos para poder ejecutar la misma. Estas actividades pueden a la vez resultar costosas al ser la transferencia uno de los criterios de facturación de los servicios en la nube.

Para analizar la combinación entre un esquema embebido y uno de nube se plantean tres aspectos: 1) arquitectura, 2) control de flujo y 3) optimización de la distribución.

### **Arquitectura**

En la mayoría de las soluciones BPM, el motor de procesos, las actividades y los datos del proceso se localizan en el mismo lado, tanto en un sistema embebido como en la nube. Existen escritos que introducen el modelo Proceso-Actividad-Datos (PAD) de la Figura 8 como una posibilidad de distribución de BPM en la nube. En el mismo, el motor de procesos, las actividades involucradas en el proceso y sus datos están separadamente distribuidos.

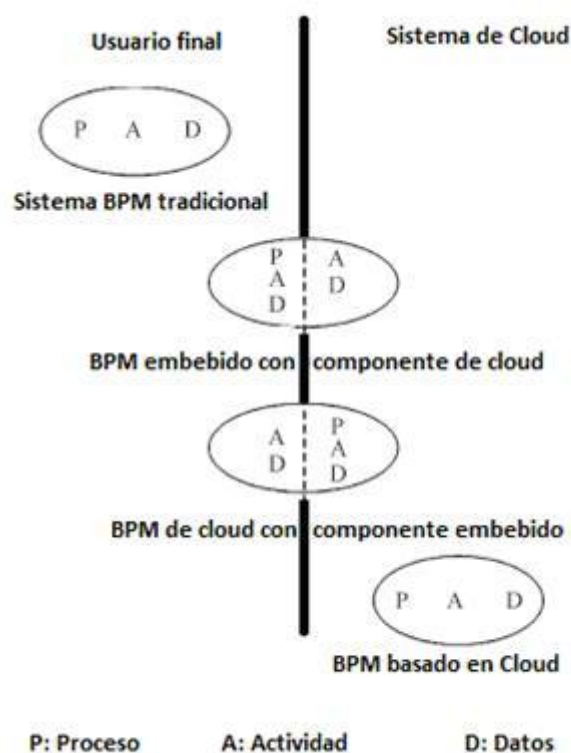
Como se muestra en la Figura 8, el modelo PAD define cuatro posibilidades de distribución:

- 1) El primer patrón de solución es el enfoque tradicional de BPM donde todos los elementos están distribuido en el usuario final (embebido).
- 2) El segundo patrón es útil cuando el usuario ya tiene un sistema BPM, pero las actividades con intensidad de cómputo se localizan en la nube para incrementar su performance.
- 3) El tercer patrón es útil para los usuarios que aún no poseen un sistema BPM, así pueden usar el esquema de nube de manera de pago por uso, y las actividades sin intensidad de cómputo y los datos sensibles se pueden localizar en el usuario final.
- 4) El cuarto patrón es el modelo basado en la nube donde todos los elementos se localizan en la nube.

### **Control de flujo**

Los procesos de negocio consisten de dos tipos de flujos: de control y de datos. Los flujos de control regulan las actividades que se ejecutan y la secuencia de las mismas; mientras que los flujos de datos determinan cómo estos se transfieren de una actividad a la otra dentro del proceso. Los motores de BPM controlan ambos flujos. Un flujo de datos puede contener datos sensibles, por lo tanto, cuando se despliega un motor de BPM en la nube, se debe proteger el contenido de los mismos.

Un ejemplo de arquitectura propuesta sería aquella en que el motor del lado de la nube solo controla flujos de datos usando identificadores de referencia en vez de datos reales. Cuando una actividad necesita datos sensibles, la transferencia de los datos a la actividad se maneja bajo supervisión del usuario dentro de un túnel de encriptación. Los datos sensibles se almacenan en el lado del usuario final, y los datos no sensibles se almacenan en la nube. Este esquema permite que los datos sensibles no viajen indiscriminadamente a través de la web.



**Figura 8. Esquema PAD de Distribución [28]**

### Optimización

Los costos de un sistema de nube han sido propósito de estudio en diversos artículos. Existen distintas fórmulas para calcular la distribución óptima de las actividades, donde las mismas pueden ubicarse en la nube o en un sistema embebido. El cálculo toma en cuenta los costos de tiempo, los costos monetarios y los costos por el riesgo de privacidad. Mediante el uso de estas fórmulas los usuarios pueden hacer una estimación de los costos de desplegar partes de sus aplicaciones en un sistema embebido y en la nube. Debemos recordar que un criterio clave desde el punto de vista económico en un sistema de nube es la tasa de transferencia de datos desde y hacia el servidor [26] [28] [29] [30].

#### 2.5.4. Descomposición de Procesos

Evidentemente, el modelo PAD posee ventajas y desventajas que no logran balancearse. Es posible generalizar la distribución e identificar un quinto patrón en el cual el motor de procesos, las actividades y los datos se despliegan en la nube y en el usuario final. Esta solución presenta dos beneficios potenciales:

El primer beneficio es que el motor de procesos regula el flujo de control y el flujo de datos. Una actividad recibe datos del motor de procesos y luego de su ejecución los datos que son producidos se pasan de nuevo al motor de procesos. Consideremos ahora que una secuencia de actividades se ubica en la nube, mientras que el motor de procesos se despliega en el usuario final. Cada actividad utiliza los datos de salida de la actividad previa como entrada. Los datos no se pasan directamente de una actividad a la otra sino que son enviados

al motor de proceso primero. Debido a que la transferencia de datos es uno de los factores de facturación en el modelo en cuestión, estas situaciones pueden volverse más caras cuando se transfieren grandes cantidades de datos entre actividades. Para evitar este problema se puede agregar un motor de procesos a la nube, el cual regula el flujo de control y el flujo de datos entre las actividades situadas en él. Cuando una secuencia de actividades se localiza en la nube, los datos se regulan por el motor de procesos en la nube, lo cual reduce la cantidad de datos a ser transferidos entre este y el sistema embebido.

El segundo beneficio es que cuando la nube no se encuentra accesible, los usuarios pueden ejecutar los procesos de negocio en forma completa en el sistema embebido hasta que el primero vuelva a estar disponible.

Para poder correr un mismo proceso de negocio en dos motores de proceso separados, el mismo debe ser dividido en dos procesos individuales. Puede llegar a ser conveniente para los usuarios del BPMS tomar una lista de distribución del proceso de negocio y sus actividades, la cual puede ser automáticamente transformada en dos procesos de negocio, uno en la nube y otro en el sistema embebido. La comunicación entre ambos sistemas puede ser descrita mediante un lenguaje de coreografías, como el Business Process Execution Language (BPEL) [31] [32].

En este tipo de modelo, el monitoreo del proceso de negocio resulta más complicado, debido a que el mismo ha sido subdividido en dos partes. Como una solución, se puede desarrollar una herramienta de monitoreo para el proceso original, mediante la combinación de los detalles de monitoreo de los procesos individuales [33] [34] [35].

Una aproximación posible para manejar la descomposición del proceso es identificar la estructura y la semántica del mismo. Cuando se identifican las dependencias de control y de datos, se pueden investigar las consecuencias de mover ciertas actividades del sistema embebido a la nube y viceversa. Cuando se conocen las consecuencias de la distribución de actividades, se puede crear una transformación de modelo en la cual un proceso de negocio y una lista con marcas se usan para crear dos procesos individuales, uno para la nube y otro para el usuario final. Además, se puede generar una descripción de la coreografía para describir la comunicación entre ambos procesos de negocio utilizando algún lenguaje estándar, como BPEL [8] [36] [4] [8].

## **2.6. BPM Y SOFTWARE SOCIAL**

Tal como se ha presentado en la sección 2.4.3 el software social provee una mejor integración de todos los involucrados en ciclo de vida de los procesos de negocio y ofrece nuevas posibilidades para un diseño de procesos más efectivo y flexible. En particular, el modelado de los procesos de negocio saca ventaja de las técnicas de software social permitiendo integrar y compartir el conocimiento de todos los involucrados. Asimismo, las etapas de implementación y despliegue



ampliadas con conceptos de software social permiten recolectar e incorporar información valiosa para la mejora continua del proceso, además de que los mismos entornos de software social pueden ser utilizados para soportar el *workflow* [4].

Sin embargo, el uso de software social implica nuevas consideraciones acerca de la identidad digital y la reputación en los procesos de negocio, siendo esta una característica que lo distingue del software social puro, donde se persigue soportar tareas colaborativas sin implicancia directa en el objetivo de negocio de una organización.

En esta sección detallamos el impacto del software social tanto en las fases de análisis, diseño, y modelado, y de implementación y despliegue de procesos de negocio. Asimismo, discutimos conceptos de identidad digital y reputación.

### 2.6.1. El Software Social en la Fase de Diseño de Procesos de Negocio

La incorporación de conceptos de software social en el diseño de los procesos de negocio, no solamente facilita la personalización de los bienes y servicios producidos por el proceso sino que las excepciones pueden transformarse en reglas, rompiendo la estructura rígida presente en el BPM tradicional.

Algunas de las consideraciones al tener en cuenta para sacar provecho de los conceptos de software social en el diseño de procesos de negocio son:

- **Socializar procesos** – En muchos BPMSs, los usuarios cuentan con una visión limitada de los procesos en los que participan, perdiendo información de contexto como las personas involucradas en el mismo y las historias y resultados de ejecuciones anteriores.
- **Atomizar las actividades** – En muchos diseños de procesos, las actividades se definen con una granularidad muy baja, lo que implica un esfuerzo sustancial por parte de los participantes para completarla. Aumentar dicha granularidad y en consecuencia atomizando las actividades, alienta la participación de los actores y permite redefinir circuitos y dividir tareas.
- **Utilizar un esquema de recompensas** – Muchos procesos organizacionales definen sus actividades de manera vertical y los participantes las ejecutan solo para cumplir con sus superiores. En el software social, la participación es voluntaria y el compromiso se establece obteniendo puntos de recompensa que pueden luego traducirse en reconocimientos informales o reembolsos monetarios pero que incluyen al participante de manera participativa e incluso lo alienta a mejorar el proceso.

En consecuencia, no se trata solamente de incluir mejores prácticas sino de mejorar las herramientas tanto en torno a BPMSs como a software social.

### 2.6.2. El Software Social en Despliegue de Procesos de Negocio

La fase de implementación y despliegue de los procesos de negocio, resulta ser la primera de las fases en la cual los aspectos tecnológicos toman protagonismo. Cuando nos referimos a procesos ejecutables, estamos definiendo el concepto de motor (computacional) que pueda poner en acción lo que se modeló en papel.

Un proceso de negocios ejecutable, se instancia varias veces y cada una de dichas instancias deja rastros y provoca la verdadera interacción con los participantes del proceso. Es aquí entonces donde cobra aún más sentido hablar de software social, dado que los conceptos subyacentes se implementan a partir de herramientas específicas y habrá que definir de qué manera se integran con las tecnologías habilitantes de BPM.

Es un hecho que en la industria del software actual, no ha habido demasiados avances en cuanto a BPMS que incorporen características sociales. Los BPMS actuales mantienen su particularidad de constituirse como motores de ejecución de procesos con muy poca capacidad para absorber interacciones con el entorno e incluso para dar soporte a procesos de negocio dinámicos o menos estructurados. En este sentido, el avance que pueda obtenerse, será útil en tanto se incorporen capacidades de integración a través de plug-ins, APIs o conectores o también con propuestas de BPMSs que incorporen estas ideas.

### **2.6.3. Identidad Digital y Reputación en los Procesos de Negocio**

Hay muchos factores de motivación para la integración del software social como parte de las prácticas de una organización y sin duda el beneficio es doble: habilita la oportunidad de mejorar los procesos de negocio a través de interacciones socialmente habilitadas y permiten crear mecanismos para que los involucrados agreguen valor al conocimiento de la empresa.

Considerando estos beneficios dentro de una organización gestionada por procesos, los mismos serán tales en la medida en que exista la identidad digital y la reputación. Ambos conceptos se explican a continuación.

**Identidad digital.** La identidad de quien gestiona la información en un sistema es aplicable tanto en el software social, como en una arquitectura orientada a servicios, una Wiki o un blog. La identificación de la identidad de quien crea los datos permite que el proceso identifique al usuario humano con la actividad que se está ejecutando.

**Confianza y reputación.** Una vez asumida la existencia de la identidad digital, es importante conocer las implicancias de la confianza y reputación. El software social requiere proveer información útil para el contexto, vinculando a los usuarios unos con otros para recabar información del entorno que alimenten a los procesos y permitan gestionar las excepciones a los procesos de negocio.

### **2.6.4. Los Procesos de Negocio y el Procesamiento de Información**

Las aplicaciones Web 2.0 ofrecen un potencial para explicar y utilizar la inteligencia colectiva existente en un proceso de negocio que si bien explicita el

conocimiento intrínseco en una organización, no considera, en su modalidad clásica, las interacciones colaborativas y globales del mundo del software social [20]. Dentro del paradigma de la Web 2.0 existe una serie de características para el intercambio de información que deben ser absorbidas por los procesos de negocio para mejorar su modelo hacia entornos colaborativos y distribuidos. Las mismas incluyen: 1) *Interacción global* – recolectar, ofrecer e intercambiar información a nivel global; 2) *Inteligencia colectiva* – intercambiar, sistematizar y evaluar información entre los participantes; 3) *Evaluación permanente* – recibir retroalimentación del comportamiento del usuario para mejorar en forma continua y permanente; y 4) *Independencia de dispositivos* – recolectar y ofrecer información independientemente de la plataforma y de la ubicación.

## 2.7. CONCLUSIONES DEL CAPÍTULO

La combinación de BPM y software social ofrece beneficios mutuos: permite que la información producida por el software social se integre a los procesos de negocio y achica aún más la brecha para proporcionar información para el diseño, implementación y mejora de los procesos, lo cual ha sido y sigue siendo uno de los principales focos de atención de BPM.

La inclusión de características de software social en las soluciones BPM, desdibuja las barreras habituales entre modelo de procesos abstractos, instancias de procesos y procesos ejecutados.

Para sacar provecho de las ventajas de los servicios de software social en BPM, se consideran un conjunto de recomendaciones fundamentadas en considerar el contexto a la hora de modelar y también modelar en función de la historia de una comunidad de usuarios. Asimismo, a la hora de desplegar y monitorear, es vital considerar trazas que la comunidad de usuarios registra en entornos Web 2.0.

## CAPÍTULO 3

### TRABAJOS RELACIONADOS

En este capítulo se realiza una revisión de trabajos relacionados con esta tesis tomando en cuenta el objetivo de mejorar y actualizar MISP [1] para obtener una solución que siga teniendo en cuenta la organización como un todo, pero revisando tanto aspectos de los modelos que la representan como de los software que los ejecutan y dejan rastros de dicha ejecución para retroalimentar la mejora continua [11] [12][13].

En este sentido el capítulo se estructura de la siguiente manera. La sección 3.1 discute los lenguajes de modelado y su vinculación con los workflows, explicando la relación entre BPM y el desarrollo dirigido por modelos. La sección 3.2 discute los trabajos relacionados a las tecnologías aplicadas; en particular, para la composición de servicios y para la disposición de procesos y servicios en la nube. La sección 3.3 explica algunas herramientas disponibles. La sección 3.4 discute el estado del arte en software social aplicado a BPM. Finalmente, la sección 3.5 presenta las conclusiones del capítulo.

#### 3.1. LENGUAJES DE MODELADO Y WORKFLOWS

Los lenguajes de modelado y su implementación a través de workflows constituyen una categoría de trabajos relacionados interesante de analizar, habida cuenta que el enfoque metodológico que se propone mejorar, aborda todas las etapas del ciclo de vida de los procesos de manera integral y donde los lenguajes para modelar los procesos constituyen el punto de partida para explicitar el conocimiento y achicar la brecha entre los involucrados. La implementación a través de workflows es un enfoque clásico para ejecutar procesos y es sin dudas la base de muchos BPMS.

En ese sentido, se analizaron trabajos donde se desarrolla esta temática para evaluar el aporte de los mismos a esta tesis.

En [57] se presenta un metamodelo basado en UML para crear y ejecutar modelos de workflow. El lenguaje de modelado se introduce a través de su sintaxis abstracta y soporta patrones de workflow ya conocidos. Implementa la herramienta UML-based Specification Environment (USE) [59] que chequea propiedades estáticas de los modelos de workflow durante el proceso de modelado observando invariantes OCL.

Por otra parte, en [58] se define una transformación entre modelos de actividad de UML 2.0 (UML-AD) a Yet Another Workflow Language (YAWL), un lenguaje formal para workflow que captura hasta 20 patrones de workflow. Este trabajo clarifica la semántica de UML-DA vía su mapeo a YAWL, simplifica el despliegue

de modelos de procesos de negocio escritos como UML-AD y además permite analizar los modelos UML-DA con herramientas de verificación en YAWL.

En la misma línea que [58], en [60] los autores de este trabajo analizan dos de los lenguajes gráficos más populares UML-AD y BPMN. Se analiza también la semántica de ambos lenguajes en el contexto de la ejecución de procesos a través de su mapeo a BPEL. Desde el análisis de los metamodelos de AD y BPMN los autores concluyen que no es trivial una transformación exacta aún seleccionando un subconjunto del lenguaje.

En [61] se aborda una implementación de Business Process Model and Notation (BPMN 2.0). Este trabajo ha resultado muy interesante en su análisis dado que al momento de la escritura de esta tesis, no existen muchas implementaciones de esta nueva versión de BPMN que soporta la inter-operación de los procesos de negocio a nivel de usuario.

En [62] se muestra el prototipo de una herramienta de modelado que aplica reglas basadas en grafos para identificar problemas en los modelos de procesos de negocio. Entre las ventajas de esta propuesta está la de poder ser aplicada a modelos de procesos de negocio incompletos. Este trabajo constituye otro ejemplo de aplicación de los lenguajes de modelados y su implementación como lenguajes ejecutables.

Finalmente en [63] se presenta una muy buena recopilación y clasificación de los lenguajes de modelado, desde los específicos para modelar procesos - como Event-driven Process Chain (EPC), redes de petri, BPMN; hasta los denominados por los autores como orientados a objetos y que utilizan distintos artefactos UML para el modelado. Asimismo analiza los que denominan lenguajes dinámicos o que aportan un motor de ejecución como Web Services Business Process Execution Language (WS-BPEL) e incluye lenguajes para interacción de procesos, como Web Service Choreography Description Language (WS-CDL). Este resumen ha resultado de mucha utilidad para validar la investigación en torno a lenguajes de modelado, realizada para esta tesis.

### **3.1.1. BPM y MDD**

La investigación realizada se centró en verificar el estado del arte en materia de propuestas metodológicas en torno a reducir la brecha entre procesos y servicios que se fundamenten en el uso de metamodelos y transformaciones entre modelos en el contexto MDD [12].

En este sentido, hemos clasificado el estudio realizado en tres grandes grupos. Por un lado, hemos encontrado diversos trabajos que aplican la técnica del metamodelado, con el objetivo de lograr versiones ejecutables de procesos de negocio a través de un workflow (Sección 3.2.1). Por otro lado, encontramos un grupo de trabajos que definen diversas reglas de transformación entre modelos (Sección 3.2.3) y demuestran la ventaja de utilizar estas reglas de transformación tanto para automatizar las transformaciones como para utilizarlas para verificar los procesos modelados. Finalmente, en la Sección 3.2.4,

agrupamos los trabajos que plantean el proceso de desarrollo de software con Model-Driven Architecture (MDA), tanto en enfoques puramente orientados a servicios como en otros donde el objetivo es disminuir la brecha entre la especificación y el modelado de procesos y servicios. Si bien los trabajos en esta categoría podrían incluirse en la Sección 3.2.1, se opta por incluirlos en una categoría separada porque su enfoque está más vinculado a la arquitectura que al uso de lenguajes de modelado.

### **3.1.2. Metamodelos para Ejecutar Procesos**

En [40] se presenta el metamodelo MOF para crear y ejecutar modelos de workflow, a través de la herramienta USE [43]. Este metamodelo permite validar propiedades estáticas de los modelos de workflow durante el modelado del proceso, observando invariantes OCL. Originado en [42], este trabajo plantea usar diagramas de clase de UML para definir un metamodelo de procesos, y diagramas de actividad (DA) para metamodelar el ciclo de vida de las actividades de los procesos. En [40], los autores permiten validar la ventaja de utilizar un enfoque dirigido por modelos como paso de interacción entre las etapas de nuestra metodología.

Por su parte en [41], se presenta el enfoque MDD en el área de definición de workflow y se muestra que el uso de un perfil UML permite ofrecer a los DA de UML como notación viable para la definición de workflow integrando a su vez datos y recursos. Una vez más, este trabajo permite confirmar que los metamodelos - propuestos como productos de cada etapa de nuestra metodología para así transformarla en un enfoque dirigido por modelos; son un componente esencial en el estado del arte actual.

Estos trabajos resultaron interesantes a la hora de validar la ventaja de utilizar un enfoque dirigido por modelos, sin embargo no pudieron aplicarse a nuestra propuesta en forma completa dado que no consideran la visión integradora de la misma, siendo así un gran aporte conceptual.

### **3.1.3. Metamodelos y Reglas de Transformación**

En [46], los autores definen una transformación desde los DA de UML hacia Yet YAWL. Si bien en este trabajo se presenta a YAWL como contrapunto de BPEL, la diferencia encontrada según nuestro trabajo da cuenta de la validez de pensar en la transformación de modelos como mecanismo útil a la hora de tender un puente entre lenguajes de diferentes niveles de abstracción y formalidad.

En [47], se presenta un prototipo de una herramienta de modelado que aplica reglas basadas en grafos para identificar problemas en modelos de procesos de negocio. El lenguaje gráfico utilizado para modelar los procesos es EPC y sobre el mismo se elabora una técnica que puede aplicarse incluso a modelos de procesos de negocio incompletos. Si bien la propuesta utiliza un lenguaje particular de modelado, los principios subyacentes pueden ser aplicados también a BPMN. Este trabajo aporta a nuestra propuesta la visión de contar con una validación de modelos en etapas tempranas del desarrollo del proyecto.

En [48] se especifica e implementa un conjunto de reglas de mapeo conceptuales entre los metamodelos de BPMN [45] y SCA [44]. Este trabajo constituye un valioso aporte a nuestra propuesta, dado que no solamente utiliza metamodelos y define reglas de transformación, sino que también utiliza los lenguajes sobre los que nos basamos. La diferencia sustancial radica en que nuestra metodología integradora propone una etapa intermedia entre el modelado de procesos en BPMN y el modelado de componentes en SCA, definiendo así una conceptualización de servicios.

#### **3.1.4. MDA como Método de Desarrollo de Software**

En [49] y [52] se presenta el método de desarrollo de software orientado a servicios Service-Oriented Development Method (SOD-M), que constituye un método dirigido por modelos para el desarrollo de servicios; en particular, Servicios web. Los procesos de negocio representan el Computational-Independent Model (CIM); la visión del sistema de información se representa primero por el modelo independiente de la plataforma (PIM) y luego se especifica en el modelo específico de la plataforma (PSM) con SoaML [51]. Los autores comparan a su vez su propia propuesta con otras orientadas a la ingeniería de software como Service Oriented Architecture Modeling Language (SoaML) [51] y Service Oriented Modeling Architecture (SOMA) [50]. Esto convalida nuestra idea de que el trabajo constituye un aporte desde el enfoque de desarrollo de software mientras que nuestra propuesta presenta una visión integradora de todo el ciclo de vida de un proyecto gestionado por procesos.

El trabajo presentado en [50], desarrolla técnicas y herramientas para dar soporte al ciclo de vida de los servicios. Se presenta un modelo para alinear modelos de procesos con especificaciones de workflow. La principal ventaja de este enfoque es la creación de un modelo de mapeo entre la realidad del negocio y la de Tecnologías de la Información llamado Business-IT-Mapping Model (BIMM), para evitar la definición de reglas de mapeo complejas. Este trabajo convalida nuestra visión.

En [53], se presenta un enfoque dirigido por modelos para tender un puente entre los requerimientos del negocio y los sistemas multi-agentes, considerando el uso de agentes en contraste a enfoques basados en motores BPEL. El trabajo se basa en los servicios de modelado a nivel PIM con SoaML y su conexión a los modelos de agentes a nivel PSM. La investigación se fundamenta en el proyecto Semantically-enabled Heterogeneous Service Architecture and Platforms Engineering (SHAPE). Este trabajo representa un aporte al enfoque dirigido por modelos e influye en nuestra propuesta ya que reduce la brecha entre procesos y servicios, pero considera a los servicios desde el modelo de agentes.

Continuando con el enfoque MDA y aplicado a la derivación de código BPEL como orquestación de servicios web, los autores de [54] proponen un nuevo método para componer servicios web usando escenarios UML especificados a través de diagramas de secuencias. El método propuesto genera el código para servicios compuestos en BPEL.

Para comprender en profundidad, el meta-modelo de WSDL [55] propone una extensión al WSDL estándar para integrar perfiles de características sobre la descripción de servicios web.

En [56] se presenta un proceso de desarrollo para interfaces de aplicaciones Web. El proceso combina SOA y SOD-M para generar interfaces de servicios web en WSDL y WS-BPEL.

En comparación con el trabajo iniciado en [1] y las mejoras planteadas en la presente tesis, [54] usa MDA para derivar BPEL desde los servicios web utilizando un enfoque en reversa (desde servicios web hacia procesos de negocio). Si bien [56] sigue una dirección similar, se utiliza solamente para interfaces de aplicaciones Web.

## **3.2. TECNOLOGÍAS**

Para analizar las tecnologías asociadas con los temas abordados por esta tesis, hemos de considerar que las aplicaciones construidas con BPM no son diferentes a las que se construyen escribiendo código, la diferencia es la capacidad de otorgar escalabilidad a la solución. El avance está dado por el trabajo colaborativo de los actores intervinientes, la modularidad de los componentes de la aplicación y la habilidad para fijar el conocimiento previo que se requiere para aplicar cambios en el medio de un proceso.

Desde esta perspectiva, las tecnologías más vinculadas a los procesos de negocio y su despliegue en entornos productivos, que encuentran puntos de contacto con este trabajo, son los servicios comprendidos como tareas computacionales débilmente acopladas que son consumidos por los procesos, la composición de los mismos a través de componentes y su composición en la nube.

### **3.2.1. Servicios y su Composición**

Un servicio web es un elemento que se comprende en términos de la utilidad que brinda, por lo tanto, no puede apartarse del problema para el cual es útil. Un servicio captura funcionalidad con un valor de negocio, y está listo para ser usado. Es provisto por uno o varios servidores, para lo cual requiere de una descripción que pueda ser accedida y entendida por potenciales clientes.

El conjunto de conceptos que describen los servicios son: su descripción - incluye la información que el consumidor requiere para usar el servicio; contratos y normas - representa el acuerdo entre las partes, condiciones de uso y restricciones; y contexto de ejecución - elementos técnicos y de negocio que habilitan su uso.

Los servicios web se basan en SOA para desplegar su arquitectura, esto permite el desarrollo de aplicaciones débilmente acopladas, las cuales pueden ser accedidas a través de la red. Por otra parte, al existir una baja cohesión entre los servicios, se hace imprescindible contar con un marco de referencia o



arquitectura, que facilite su registro y publicación para conocer su existencia [64] [65]. Si bien los servicios web surgieron en forma paralela a la idea de SOA, esta arquitectura facilitó y ordenó el uso sistemático de los mismos combinando protocolos, perfiles, especificaciones y estándares [66] [67].

En torno a estas ideas, [68] presenta un buen análisis de las exigencias de implementación que presenta SOA y el cambio de pensamiento que se requiere en las etapas de diseño, desarrollo y despliegue cuando nos involucramos con BPM. Este trabajo analiza el cambio y la propagación de la automatización así como el desacoplamiento y la dosificación de los procesos monolíticos a los servicios individuales, analizando la influencia de estos cambios en las personas.

Continuando con el análisis de tecnologías de servicios, en [69] se propone un modelo de diseño de procesos en lenguaje BPEL que admite la selección dinámica de servicios web. Además, los procesos BPEL generados por el método pueden desplegarse en cualquier motor de ejecución BPEL. Esta propuesta analiza pormenorizadamente la problemática planteada en torno a las características estáticas de los procesos BPEL que también pueden trasladarse a los procesos de negocio modelados y ejecutados en otras tecnologías.

El despliegue de servicios web, así como su orquestación y composición con las tecnologías asociadas, también ha sido estudiado en torno a los métodos para testear su funcionamiento, como una temática natural del desarrollo de software.

Así en [70] se plantea que si bien los servicios individuales pueden funcionar correctamente, varias fallas inesperadas pueden provocarse durante la ejecución de servicios web compuestos, siendo difícil detectar la falla original debido a que las fallas pueden propagarse y acumularse. La solución propuesta crea un modelo de prueba desde los diagramas de secuencia de UML y derivan un camino de pruebas desde el modelo de servicios web compuestos.

Por su parte, [71] presenta un enfoque para testear servicios web de caja negra basado en eventos, que facilita tratar con servicios web bajo circunstancias regulares (es decir, cooperando como se esperaba), así como bajo circunstancias no habituales (con servicios web cooperando de manera inesperada). Asimismo la propuesta puede utilizarse independientemente de la plataforma de orquestación (por ej, BPEL) y del tipo de composición en sí mismo (orquestación o coreografía). Este trabajo muestra la complejidad existente a la hora de testear servicios no individuales debido a sus características de funcionamiento y a la necesidad de adherir a los flujos fijados por sus orquestadores.

Otro aspecto vinculado a los servicios y su despliegue que ocupan el interés de los investigadores, está relacionado con la identificación de los servicios. Por ejemplo, en [72], se presenta un enfoque global e integrado de los métodos existentes de identificación de servicios. Los autores consideran que un factor clave que determina si una organización se puede beneficiar con la adopción de SOA es el diseño de los servicios y dentro de ellos, la identificación de los mismos. Estos métodos varían desde la extracción de código fuente para analizar el dominio del negocio, pasando por estrategias bottom-up y top-down y

alcanzando también el uso de ontologías. Este trabajo incluye un reporte y clasificación de 30 trabajos vinculados a las metodologías para identificar servicios y que han sido de utilidad a la hora de revisar la fase de modelado de servicios propuesta en [1] para analizar mejoras de interacción entre dicha etapa y la subsiguiente de modelado de componentes. En particular, convalida el enfoque propuesto en [12] a partir de la definición de un metamodelo que articule con el metamodelo de SCA.

Continuando esta línea de análisis, en [73] se aborda la problemática de adoptar SOA para pequeñas y medianas empresas, donde los recursos, los tiempos y el aspecto financiero presentan un escenario incompatible con los costos de tal adopción. En este sentido, el trabajo plantea evaluar la cantidad y variedad de entradas consideradas por los métodos de identificación de servicios para alcanzar un método que considere la situación de las pequeñas y medianas empresas. Dentro de los tipos de entradas para los métodos de identificación de servicios se pueden considerar: los procesos de negocio, los casos de uso, los diagramas de actividad, el código fuente, las interfaces de usuario, las bases de datos, los objetivos organizacionales y el conocimiento del dominio del problema. Este artículo contiene un detallado análisis de todos estos tipos de entradas y su aplicación en la identificación de servicios, permitiendo derivar los más adecuados para aplicar en el contexto de las pequeñas y medianas empresas.

Por último, en [74] se presenta una plataforma que incluye herramientas y técnicas que permite integrar la composición de servicios en un entorno distribuido y considerando calidad de servicio. Esto último es logrado mediante un lenguaje que permite la inclusión de requisitos de calidad de servicio y parámetros relacionados con su ejecución incluidos en la descripción del mismo. Esta propuesta, si bien presenta un Framework en particular, da cuenta de la complejidad existente para desarrollar este tipo de entornos y además lo vincula a un lenguaje y máquina de ejecución propios. En el marco de esta tesis, este trabajo resultó de interés a la hora de dimensionar la complejidad tecnológica que implica el despliegue de servicios pero no ha aportado un enfoque metodológico que ayude a mejorar las etapas de la metodología integradora de procesos y servicios propuesta en [1].

### **3.2.2. Disposición de Procesos y Servicios en la Nube**

En [75] se detalla el modelo de computación en la nube y su tecnología, así como también el despliegue de un entorno de BPM ubicando en cada modelo de servicios las distintas componentes de un BPMS. Este trabajo es una puesta en escena inicial acerca de las futuras posibilidades que brinda la nube al paradigma BPM. No aborda cuestiones de distribución de procesos y tampoco de un monitoreo distribuido como es el foco de nuestro trabajo.

Por otra parte en [76] se aborda el impacto de la nube en los BPMS como un salto en la escala de madurez en la adopción de BPMS, considerando la distribución de sus componentes bajo cuatro esquemas diferentes y analizando su aplicación acorde al dominio del problema. Asimismo refiere el concepto de iBPMS (*iBPMS – Intelligent Business Process Management*) como una evolución

de dicha tecnología hacia entornos sociales, colaborativos y por ende, vinculados a la nube. Sin embargo no plantea un escenario concreto de distribución de procesos ni analiza peculiaridades de cumplir con aspectos importantes de la mejora continua de los procesos mediante el análisis de un monitoreo de procesos distribuidos.

En [77], encontramos una muy buena propuesta de distribución de procesos que de hecho se tomó como base para una alternativa propia donde se combinen diferentes estrategias con el fin de definir una arquitectura de monitoreo distribuido (tal como se propone en [145]).

A su vez en [78] se presenta una extensión a los BPMS aportando la capacidad de llevar el rastro de eventos, además de actividades utilizando la idea de Complex Event Processing (CEP). Este enfoque trata el tema de monitoreo de servicios en profundidad mediante el mismo mecanismo propuesto por nuestro trabajo: el de enriquecer los rastros que dejan las ejecuciones de los procesos; sin embargo, no aborda el enfoque de manera distribuida.

En [79], se plantea un mecanismo de monitoreo enriquecido por OCL desde una visión centralizada. Este trabajo ha sido fuente de inspiración para elaborar una propuesta distribuida sobre un caso real.

En [80] se propone una arquitectura BPM con distribución en el usuario final y en la nube según las actividades requieran capacidad computacional o acceso a datos sensibles. Se prueba experimentalmente que el enfoque puede sacar ventajas de este esquema de distribución y constituye una opción de distribución híbrida que valida el punto de vista abordado en esta tesis.

Por su parte en [81] se plantea la descentralización de procesos como paso hacia soluciones empresariales a gran escala. Se propone una metodología para transformar una especificación de procesos centralizada en una ejecución distribuida que incorpore sincronismo entre las entidades procesadas.

En el análisis de los workflows presentados como SaaS, encontramos en [82] una descripción de cómo las componentes existentes en tecnología workflow pueden integrarse y extenderse a una plataforma basada en la nube. En esa misma línea en [83] se presenta una solución que planifica en forma automática los pasos de un workflow para host con baja carga de uso y provee nuevos host usando infraestructura de nube en situaciones de carga máxima. Asimismo, [84] analiza en profundidad la tecnología de computación en la nube y workflow discutiendo el escenario de aplicación y casos de workflow en la nube que proporcionan definición abstracta de usuarios, configuración flexible, operación automatizada de aplicaciones complejas, programación de tareas y gestión de recursos en la nube.

Continuando con el análisis de tecnologías vinculadas a la ejecución de procesos en la nube, encontramos en [85] una descripción acerca de cómo afectan la seguridad y la confiabilidad en la ejecución de procesos BPEL en la nube y muestra los requerimientos de middleware para ejecutar tales procesos. La investigación se realiza tanto en modelo IaaS como SaaS. En la misma línea,

en [86] se presenta una plataforma de BPM que asiste a los analistas de negocio sin experiencia en programación para automatizar pasos manuales y otorga una guía de recomendaciones para diseñar, implementar, desplegar y ejecutar procesos de negocio en un entorno de nube híbrido.

Finalmente, abordando el tema de monitoreo distribuido para los procesos de negocio se analizaron dos trabajos, uno de ellos relacionado con el monitoreo distribuido en general y otro vinculado al monitoreo de coreografía de servicios que han aportado modelos conceptuales importantes. En particular, en [87], se plantea la problemática en torno a la dificultad de monitorear la infraestructura subyacente presente en la nube y existen regulaciones que deben cumplirse, más que nada cuando se atraviesan las fronteras de los países. El trabajo presenta una arquitectura de *login* para modelos IaaS que facilita el monitoreo de datos del usuario que no sólo muestra cómo están siendo usados sino que determina la conformidad con los requerimientos, utilizando descriptores eXtensible Markup Language (XML). La arquitectura permite llevar el rastro completo de las operaciones en la nube. Finalmente, [88] presenta un método general para monitorear coreografías de procesos basados en una instancia de proceso unificada mediante el uso de servicios web, así puede ser integrada a un entorno de ejecución de procesos.

### 3.3. HERRAMIENTAS

En [1], se presenta un estudio de las herramientas disponibles para dar soporte al ciclo de vida de los procesos de negocio y se definen las características más importantes que deben cumplir dichas herramientas. Asimismo en [89], se presenta una matriz de evaluación de herramientas y sus componentes. Este enfoque mantiene vigencia dado que las herramientas han madurado en cuanto a prestaciones pero mantienen la arquitectura básica, no otorgando demasiadas variantes, tanto en sus versiones propietarias como de código abierto. Sin embargo, el aspecto más relevante que se encuentra en la evolución de la tecnología se encuentra ligado a su posibilidad de despliegue en la nube. Como se explicó en la Sección 2, para esto, existen modelos diferentes relacionados con el modelo de despliegue y el modelo de servicio.

Considerando que el modelo de despliegue es un aspecto netamente de infraestructura que excede los alcances de este trabajo, nos concentraremos en los diferentes modelos de servicios, lo que nos llevará a comprender, en términos de herramientas de software, cuales son las variantes y antecedentes de las mismas en torno a BPM y servicios.

Desde el punto de vista de la ingeniería de software, en términos de la nube se habla de SaaS, como el modelo de provisión de software donde no se requiere instalación del software por parte del usuario, ni equipamiento de soporte - ya que más de un cliente puede acceder al software, y donde el modelo de distribución y pago se realiza por suscripción. Este modelo no difiere, de una aplicación Web tradicional, salvo por ciertas características que enunciaremos a continuación:

### **1) Debe ser multi-tenant**

El concepto de *multi-tenant* o multi-inquilino refiere a un principio de arquitectura de software en donde una única instancia de un producto de software corre en un servidor, atendiendo a múltiples organizaciones o clientes. Cada cliente u organización dentro de la misma instancia de la aplicación tiene su propio ambiente o partición, es decir, puede personalizar la aplicación definiendo sus propios usuarios, mecanismos de seguridad, parámetros y configuraciones visuales sin interferir a las otras organizaciones y de forma totalmente transparente. El objetivo primario de la arquitectura *multi-tenant* es maximizar recursos de hardware y de software. La arquitectura *multi-tenant* se opone de alguna forma a la arquitectura multi instancia, en donde para cada cliente u organización se necesita instalar una nueva instancia de la aplicación con sus respectivos recursos de hardware y software dedicados para cada organización.

### **2) Debe ser escalable y proveer mecanismos de balanceo de carga**

Las aplicaciones SaaS deben estar preparadas para soportar una gran cantidad de clientes. En aplicaciones *single-tenant* la escalabilidad se logra instalando un nuevo servidor web con la misma aplicación y balanceando la carga. Eso para soluciones *multi-tenant* es una solución de granularidad grueso, por lo que es necesario definir mecanismos de escalabilidad que contemplen el uso de recursos por cliente y la posibilidad de asignar recursos de grano fino.

### **3) Debe ser personalizable y configurable**

Cada cliente que se suscribe al servicio, utiliza la aplicación como si fuera el único cliente de la misma, por lo que la aplicación necesita diseñarse de tal forma que permita que cada inquilino pueda personalizarla de acuerdo a sus necesidades sin interferir a otros.

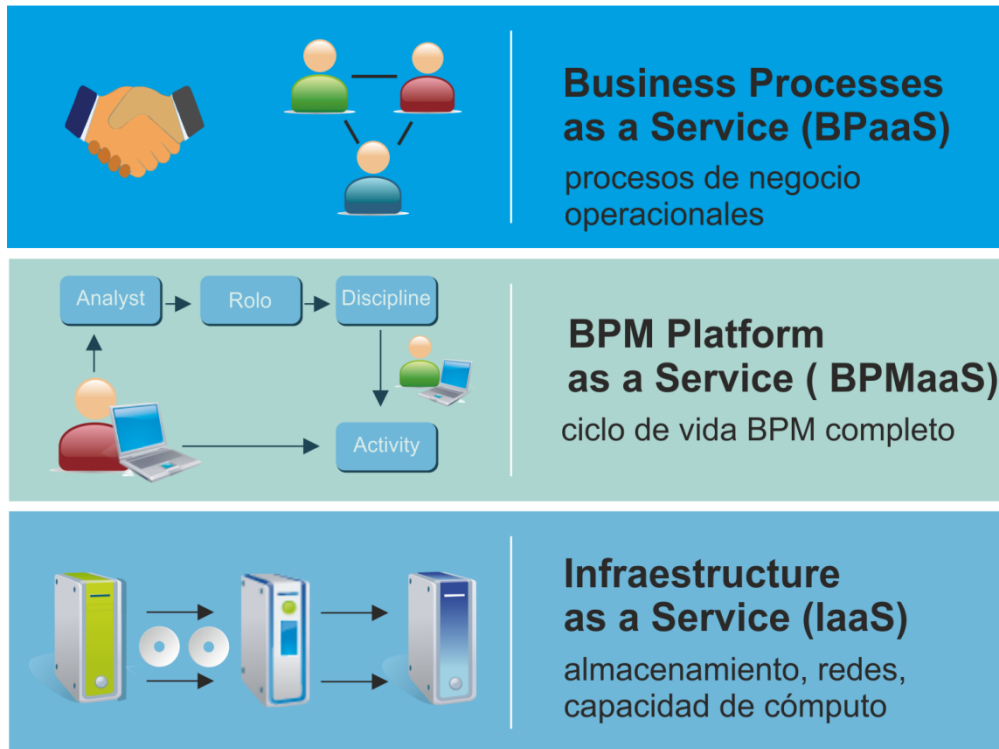
### **4) Debe proveer mecanismos de suscripción, monitoreo, monetización y facturación**

El modelo de pago por uso implica que en la aplicación se tenga que diseñar específicamente los mecanismos de suscripción, monetización y facturación, por lo que se necesita monitorear constantemente el uso que cada inquilino le da a la misma. Es prioritario ofrecer al cliente una buena variedad de precios, y que el inquilino sepa exactamente qué se le está cobrando.

En este sentido, entendiendo BPM como SaaS, lo que obtendríamos serían soluciones gestionadas por procesos, es decir, el despliegue propio de los procesos en un entorno de nube y que los usuarios ejecuten sus procesos en la nube. Esta variante, es la que se denomina BPaaS, donde los procesos en ejecución son utilizados por clientes, proveedores, empleados y diversos participantes para ejecutar sus tareas. El proceso se despliega sobre una infraestructura de nube y es accesible vía un teléfono inteligente o una tableta.

Otra consideración a realizar en términos de la nube y BPM es la visión bajo el modelo de despliegue PaaS, donde se provee al usuario de los medios para desplegar y gestionar los procesos de negocio usando herramientas BPMS

soportada por servicios en la nube. El usuario no controla la infraestructura pero gestiona los procesos desplegados como BPaaS bajo el modelo Business Process Management as a Service (BPMaaS). Con BPMaaS se pueden cargar servicios en lugar de aplicaciones directamente desde la plataforma incluso con la variante “bajo demanda” permitiendo escalar horizontal y verticalmente. Inspirados por [90], la Figura 9 muestra lo descrito en esta sección.



**Figura 9. Tres Tipos de Modelos de Despliegue de Servicios**

### 3.4. ESTADO DEL ARTE EN SOFTWARE SOCIAL APLICADO A BPM

Entre los trabajos analizados a la hora de formular esta propuesta, se ha notado que se trata de un área de investigación muy incipiente donde se están dando los primeros pasos en la construcción de propuestas, elaboración de técnicas y construcción de plataformas. Sin embargo, se han encontrado cuatro trabajos que permiten fundamentar la propuesta presentada en esta tesis y que han contribuido a la formación de conceptos para su desarrollo en el prototipo.

En [4] se analiza en profundidad al software social como mecanismo para mejorar la integración de los interesados tanto en la fase de diseño de procesos como en las fases de implementación y despliegue, recolectando información valiosa para la mejora continua. Este trabajo constituye un buen fundamento a la propuesta presentada en [4] y convalida el prototipo presentado.

En [23] se analiza también la integración entre BPM y software social con vistas a proponer un metamodelo dirigido a objetivos para las etapas de diseño y

promulgación de los procesos. El enfoque propone el descubrimiento de procesos pero no presenta aún un caso de estudio concreto. Asimismo, los autores afirman que se trata de un área de trabajo muy poco explorada aún. Este trabajo fundamentó nuestra propuesta aportando importantes definiciones.

En [15] se plantea el enfoque de BPM ágil como la capacidad de BPM de reaccionar rápidamente a los cambios y cómo lograr poner al ciclo de vida de los procesos bajo un nuevo paradigma donde se apliquen características de software social. La propuesta plantea el concepto de procesos de negocio sensibles al conocimiento (knowledge-sensitive) que no puede representarse en forma rígida y que impone un cambio en las notaciones y en los lenguajes de procesos de negocio que favorezcan la operación y coordinación. Este trabajo muestra cómo usar el software social para afrontar los desafíos de BPM ágil, pero no desarrolla en forma completa el ciclo de vida ágil. El aporte para nuestra propuesta se basa en el análisis de un ciclo de vida ágil de los procesos, que incorpora eventos internos y externos integrando a todos los involucrados.

En [19] se plantea como mejorar la gestión de procesos de negocio monolítica a la luz de la Web 2.0 y se lleva a cabo este concepto con una plataforma colaborativa de BPM que integra el modelado colaborativo con el uso de inteligencia colectiva. Los autores proponen integrar herramientas de trabajo colaborativo con estrategias de tagging (agregado de etiquetas) para el modelado de procesos colaborativo y la recolección de inteligencia colectiva mediante etiquetas. Estos conceptos vinculados al paradigma 2.0 convalidan la propuesta de prototipo presentada en este trabajo y que abarca la fase de ejecución y monitoreo de procesos, en lugar del modelado y análisis.

### **3.5. CONCLUSIONES**

En esta sección se han presentado los trabajos relacionados a los temas abordados en esta tesis y se realizó una clasificación del estado del arte en torno a metodologías, tecnologías y herramientas que se encuentran vinculadas a MISP que se propone actualizar, mejorar y formalizar.

De acuerdo al análisis realizado sobre los trabajos relacionados se observa en general una ausencia de propuestas integradas que tengan en cuenta todo el ciclo de vida de los procesos de negocio. Asimismo, las propuestas de integración existentes se encuentran circunscriptas a tecnologías y frameworks específico que resulta dificultoso de generalizar.

Por otra parte, se observa una ausencia de propuestas para el modelado de servicios como elementos que realizan las actividades de los procesos. Esta falta de integración favorece a que los modelos no se puedan mantener actualizados y ensanchan la brecha entre los procesos y su implementación.

En relación con el análisis realizado sobre aspectos más tecnológicos como la computación en la nube y el software social, se encuentran hasta el momento de publicación de este trabajo, pocas propuestas que consideren estos aspectos como parte de los motores de ejecución de procesos.

Finalmente, los aspectos de mejora en torno a la ejecución de procesos como lo es un monitoreo enriquecido, no encuentran por el momento un abordaje completo en los trabajos analizados.



## CAPÍTULO 4

# METODOLOGÍA INTEGRADORA DE PROCESOS Y SERVICIOS

En este capítulo se describe la Metodología Integradora de Procesos y Servicios (MISP) junto con las mejoras introducidas para esta tesis. Además se compara la misma con otras propuestas y se presenta un ejemplo de aplicación de la misma.

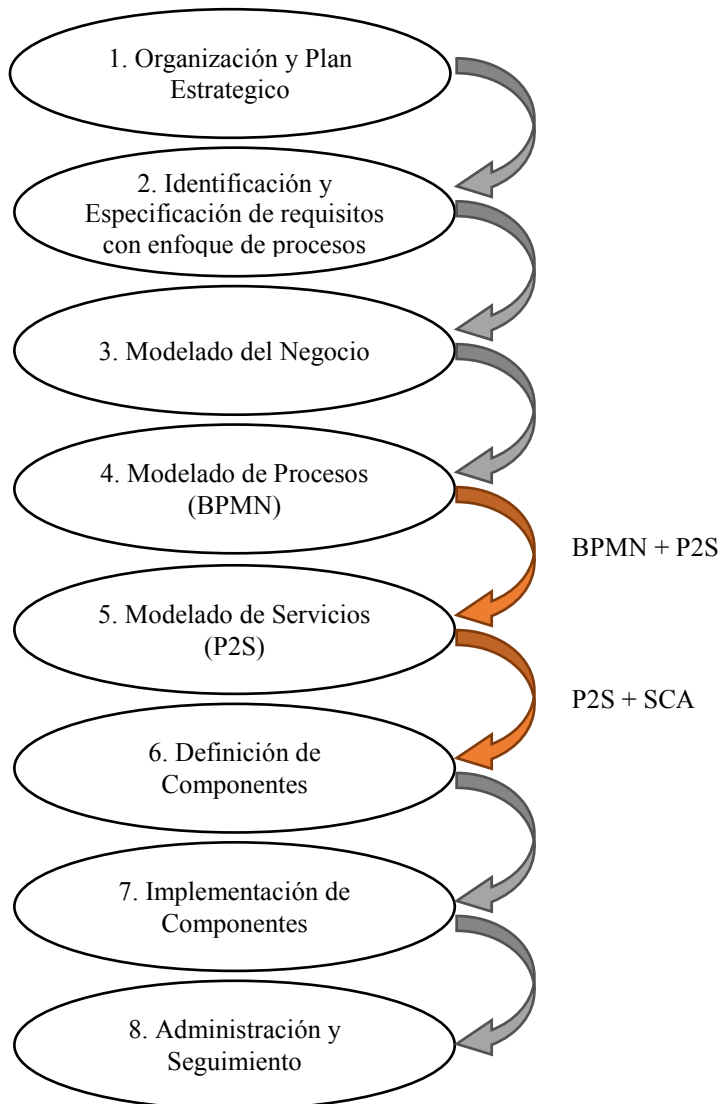
### 4.1. INTRODUCCIÓN

MISP, presentada en [1], propone un modelo para alinear los procesos de negocio con los servicios que proveen la funcionalidad requerida por aquéllos. Dentro de esta metodología, la conceptualización de servicios y su mapeo con componentes de software permite reducir la brecha entre el ciclo de vida de los procesos de negocios y sus versiones ejecutables. Si bien en la definición de la metodología se incluyeron un conjunto de etapas, la interacción entre las mismas aún debía formalizarse. El uso de metamodelos es una alternativa para esta formalización, ya que los metamodelos permiten definir la sintaxis de los lenguajes de cada etapa y las reglas de transformación entre ellos sin ambigüedades, sirviendo como paso previo a las transformaciones automáticas.

Este trabajo tiene como objetivo mejorar la formalización de cada etapa de MISP a fin de generar productos más robustos que permitan interactuar dentro del ciclo, tal como se muestra en la Figura 10. Para ello se plantea una integración de metamodelos entre BPMN y Service Component Architecture (SCA) pasando por un metamodelo propio para describir el modelo de servicios denominado Process To Service (PS2) [91]. Asimismo, se propone también una integración entre los metamodelos de SCA y WSDL como una manera de implementar componentes como servicios. Específicamente se plantea la necesidad de favorecer la interacción entre la naturaleza top-down del modelado de procesos con la característica bottom-up del modelado de servicios.

Los aportes del presente trabajo para alcanzar los objetivos son:

- 1) Producir un metamodelo propio para describir el modelo de servicios denominado P2S [91].
- 2) Plantear una integración de metamodelos entre BPMN y SCA pasando por P2S [13].
- 3) Proponer una integración entre los metamodelos de SCA y WSDL como una manera de implementar componentes como servicios. [12] [124].



**Figura 10. Etapas de MISP con Interacción Mejorada**

Este capítulo se organiza de la siguiente manera: en la sección 4.2 se describe someramente MISP cuyos detalles pueden encontrarse en [1], y se detallan más pormenorizadamente las etapas para las cuales se definen los metamodelos y sus interacciones. En la sección 4.3 se presenta una comparación de las propuestas existentes en materia de interacción entre procesos y servicios. En la sección 4.4 se introduce la metodología mejorada propuesta con una descripción de sus metamodelos y alternativas de implementación. Finalmente, la Sección 4.5 presenta un ejemplo aplicando esta mejora metodológica.

## 4.2. MISP. DESCRIPCIÓN FOCALIZADA EN LAS ETAPAS REVISADAS

MISP propone abordar la solución de problemas con una visión integradora de procesos y servicios, a través de la definición de un marco metodológico. En la definición de MISP se han considerado dos aspectos relevantes: por un lado, el carácter ortogonal de los servicios respecto de los procesos y viceversa; por el otro, la convivencia de un modelo de ciclo de vida iterativo con uno en cascada,

correspondientes a la gestión de procesos de negocio y los servicios respectivamente, que comparten características comunes. La noción de ortogonalidad de los procesos respecto de los servicios se fundamenta en el hecho de que los procesos atraviesan las áreas funcionales de la organización y alcanzan el objetivo para el que fueron definidos, como una coordinación de las funcionalidades que cada área resuelve. Los servicios, por su parte, son elementos que se comprenden por la utilidad que brindan y, por lo tanto, no pueden apartarse del problema que resuelven. Los servicios resuelven funcionalidades concretas de cada unidad funcional que cuando son coordinados dentro de una misma organización, se habla de orquestación y cuando trasciende la organización para integrarse con otra, estamos en presencia de coreografía, tal como se presentó en el Capítulo 1.

La Figura 11 presenta un ejemplo donde se muestra la ortogonalidad de los procesos respecto de los servicios dentro de una organización.



**Figura 11. Ortogonalidad de Procesos y Servicios**

El ciclo de vida de los procesos de negocio se caracteriza porque cada etapa es recorrida en forma cíclica, sin tener un orden temporal, aunque sí un orden lógico. Este ciclo puede recorrerse entrando en cualquier fase, pero una vez dentro del ciclo, se debe continuar con la próxima etapa, sin contar necesariamente con una condición de finalización. Es muy similar a un ciclo de vida en espiral. Por otro lado, el ciclo de vida de los servicios como el del software, en general tiende a tener un comportamiento en cascada. Si bien las nuevas metodologías de desarrollo de software, incluso ágiles, han marcado una evolución en la manera de construir software, su resistencia a la absorción de cambios sigue siendo una característica. Sin ir más lejos, es difícil entrar al ciclo en cualquier etapa por la sencilla razón de que dicho ciclo no está activo en forma permanente. En este contexto, la gestión de procesos de negocio (BPM) y las arquitecturas orientadas a servicios (SOA) encuentran un marco de aplicación que las integra.

La integración de BPM y SOA presenta ventajas. Por un lado, la propuesta BPM ha adquirido una atención considerable recientemente tanto por las comunidades de administración de negocios como las de ciencia de la computación. BPM es una estrategia para gestionar y mejorar el rendimiento de un negocio optimizando sus procesos a través del modelado, ejecución y medida

de rendimiento dentro de un ciclo de mejora continua [92]. Por otro lado, SOA no es un concepto nuevo, sino un enfoque diferente para el diseño y construcción de sistemas que sean flexibles y adaptables para apoyar un entorno empresarial dinámico [93]. En este marco, los servicios pueden compartirse y reutilizarse en varios procesos de negocio. El resultado es un entorno altamente adaptable, con menores costos para el desarrollo de aplicaciones, mejoras en la integración y despliegue rápido.

En [1] se presenta “Un Modelo de Integrabilidad con SOA y BPM” y se desarrolla una metodología (MISP) con el objetivo de concebir un modelo de integración de aplicaciones, dentro de una organización, de modo de alinear los procesos que definen su funcionamiento con los servicios que dan soporte a la funcionalidad. A continuación se enuncian las ocho etapas que la componen (ver Figura 10), con una breve descripción de su alcance.

- 1) **Etapas de Organización y Plan Estratégico.** Aborda el problema, contiene actividades que continuarán siendo desarrolladas durante todo el ciclo de vida del proyecto. El objetivo principal es delinear el perfil de la solución y fijar objetivos y estrategias claras que serán conservados a lo largo del proyecto.
- 2) **Etapas de Identificación y Especificación de Requisitos con enfoque de procesos.** Analiza los requisitos desde el punto de vista de los procesos. Se incluye en esta propuesta una metodología basada en la idea del Diseño Participativo de Procesos [94].
- 3) **Etapas de Modelado del Negocio.** Identifica los procesos de negocio y sus principales restricciones. Los procesos se describen como un conjunto de tareas en la que los actores participan según un flujo de trabajo determinado.
- 4) **Etapas de Modelado de Procesos.** Modela cada uno de los procesos identificados y detallados en los casos de uso del negocio mediante el lenguaje estándar de la OMG [96], para modelado de procesos del negocio, BPMN [97] [98], obteniendo un diagrama del proceso de negocio (BPD). Se aplican restricciones y objetos de información documentados mediante un formulario genérico que acompaña al BPD.
- 5) **Etapas de Modelado de Servicios.** Un servicio identifica un ámbito bien definido de funcionalidad y es accesible de forma uniforme y completa [92]. Esta etapa define el grado de granularidad de la pieza diseñada como para que se pueda considerar un servicio, con autonomía y atomicidad.
- 6) **Etapas de Definición de las Componentes.** Define componentes de software en términos de los servicios identificados y sus interacciones (orquestración).
- 7) **Etapas de Implementación de las Componentes.** Se despliegan las componentes definidas en la etapa anterior en la plataforma elegida, Incluye el desarrollo de prototipos y la integración con los sistemas existentes

8) **Etapa de Administración y Seguimiento.** Incluye las actividades necesarias para evaluar medidas de rendimiento y mejora continua.

En esta propuesta se observa la definición de una etapa de modelado de proceso y otra de modelado de servicios que constituyen el eje central de la metodología debido a su impronta integradora. Esta integración tiene por objetivo achicar la brecha que se genera entre el mundo de los procesos de negocio y el de la tecnología que los despliega, de modo que la interacción entre dichas etapas resulta central. Por esta razón se focaliza la revisión y mejora de la interacción entre: 1) procesos - modelados como paso de refinamiento desde el modelo del negocio y 2) servicios - conceptualizados para dar respuesta a las actividades de los procesos. Los servicios conceptualizados pueden ser considerados como nuevos servicios de negocios construidos para realizar las actividades de los procesos, o bien como funciones de negocios de aplicaciones existentes, que serán reusadas como servicios.

#### **4.2.1. Descripción Detallada de la Etapa 4 - Modelado de Procesos**

A partir del mapa de procesos ya construido en etapas previas, las necesidades y requisitos identificados y especificados y los casos de uso del negocio descritos en la etapa de modelado del negocio, se puede comenzar la etapa de modelado de procesos de negocio. Si bien el modelado de los procesos de negocio constituye una de las piezas fundamentales para desarrollar soluciones con enfoque en los procesos de negocios, resulta insuficiente abordar solamente el aspecto funcional de tales procesos, debiendo completarse con la perspectiva de los datos que circulan en los flujos de trabajo que orquestan las actividades de los procesos. Más aún, desde un enfoque basado en servicios, es indudable que tales servicios conformarán componentes de software donde los datos deben no solamente ser modelados sino también persistidos e intercambiados.

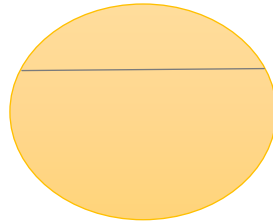
En este marco, se propone desarrollar esta etapa mediante un diagrama de procesos de negocio o Business Process Diagram (BPD) que se obtiene utilizando BPMN, pero enriquecido con la especificación que aportan dos documentos: el de los casos de uso del sistema y el de descripción de los objetos de información. Estos dos documentos se construyen con diagramas casos de uso de y diagramas de clases, respectivamente. En cuanto a los BPD pueden construirse con cualquier herramienta de modelado que soporte BPMN. Existe una gran variedad de herramientas para el modelado de procesos, muchas de código abierto que han sido analizadas en [99].

#### **4.2.2. Descripción Detallada de la Etapa 5 – Modelado de Servicios**

La idea de un servicio es identificar un ámbito bien definido de funcionalidad y hacerlo accesible de forma uniforme y completa, independientemente de dónde se use [92]. Los servicios incorporan reglas de negocios, además de información y operaciones. Este es uno de los puntos que los diferencian de los objetos.

Un servicio es un objeto con mucha más información y un menor grado de acoplamiento con otros servicios dado que no existe una asociación entre ellos,

sino que su comportamiento es desencadenado por la meta-información que posee, básicamente en cuanto a contrato de servicio y contexto de ejecución. Debido a su similitud con los objetos, admiten que sean modelados en términos de clases. Inspirado en las clases de UML, Erl define en [100] el símbolo de servicios como un círculo comprendido por dos áreas: una para el nombre del servicio y otra para las capacidades del mismo, tal como se ve en la Figura 12. El fundamento del uso de este símbolo se inspira en la idea geométrica de círculo como reflejo de autonomía, independencia e individualidad, estableciendo una unidad lógica denominada servicio [100]. Esta unidad lógica es abstracta, independiente de la implementación y es un elemento del inventario de servicios.



**Figura 12. Símbolo de Servicio como Círculo con Cuerda**

Es importante considerar la correcta granularidad de los servicios, así como su categorización para evitar lo que se denomina “síndrome de servicio”- la tendencia a generar servicios muy atomizados que pierden algunas de las propiedades, como la reusabilidad o el encapsulamiento. Si bien los servicios son los bloques constructivos de los procesos de negocios, no todos los servicios se categorizan como servicios de negocio. Por ejemplo, un servicio centralizado que informa condiciones de excepción, o un servicio que establece la conexión con una base de datos relacional, son servicios que pueden utilizarse en muchos procesos de negocio y se los denomina servicios de infraestructura o servicios de tecnología.

Los servicios de negocio abarcan los servicios de entidad, donde su alcance funcional son las entidades de negocio. Son de alta reusabilidad dado que son agnósticos a muchos procesos de negocios. El otro aspecto que abarcan los servicios de negocios son los servicios de tareas que son servicios de negocio con límites funcionales directamente asociados con una tarea o proceso de negocio. Es un servicio que tiene menos reuso potencial y es generalmente responsable de controlar la composición de otros servicios, por lo tanto, entre sus capacidades, encapsula lógica de negocio.

Los contratos de servicio indican claramente las interfaces que proveen, sus operaciones y sus parámetros, favoreciendo la catalogación de los servicios y su consecuente composición para ampliar las funcionalidades. Por último, otro aspecto muy importante es la secuencia de interacción entre servicios que se puede mostrar mediante diagramas de secuencia de UML.

La etapa de modelado e identificación de servicios implica acompañar el cambio de estado de dicho servicio desde lo intangible a lo físico, tal como se afirma en [101]. Allí el autor expresa que el modelado orientado a servicios es una disciplina guiada por la metamorfosis de los servicios, donde el curso normal comienza con la inspección de los mismos que emergen como concepto,

continúa a través de su análisis y diseño y culmina como una solución ejecutable.

### **4.3. INTEGRACIÓN DE PROCESOS Y SERVICIOS - PROPUESTAS EXISTENTES**

Entre los métodos y frameworks existentes y analizados para este trabajo, se plantea una clasificación en tres categorías: servicios, procesos y la interacción entre ellos. Para cada categoría, se describen trabajos relacionados relevantes y finalmente se establece una comparación entre los mismos y la propuesta de mejora a MISP planteada en este capítulo.

#### **4.3.1. Servicios**

Service-Oriented Modeling and Architecture (SOMA) [50] es una metodología de desarrollo de software creada por IBM. SOMA permite al usuario diseñar y construir soluciones basadas en SOA. SOMA define técnicas y provee tareas detalladas para guiar el análisis, diseño, implementación, testeado e implementación de componentes, para los servicios.

Service Oriented Architecture Modeling Language (SOAML) [51] describe un perfil de UML y un metamodelo para servicios dentro de una arquitectura orientada a servicios. Su objetivo es dar soporte a las actividades, modelando y diseñando servicios dentro de un enfoque dirigido por modelos.

e3-value [102] es una metodología usada para modelar e interconectar servicios. El método e3-value es soportado por herramientas gráficas con cierta capacidad de razonamiento y es muy adecuado para representar procesos y servicios inter-organizacionales. Su ontología permite describir organizaciones que se comportan de acuerdo a ciertos mecanismos regulatorios [103]

El principal objetivo del proyecto Eclipse SOA Tools Platform (STP) [104] es construir frameworks para mantener el ciclo de vida de los servicios en SOA. Uno de los sub-proyectos activos de STP provee un editor para BPMN [111] que se ha utilizado en este trabajo.

#### **4.3.2. Procesos**

IBM WebSphere Business Modeler [105] and Rational Modeler [106] son dos ejemplos de herramientas que asisten en la transformación de los procesos de negocios en un modelo cercano a SOA y expresado en UML.

El enfoque de procesos aplicado a la construcción de aplicaciones Web es una perspectiva enfocada en [39] [108] a través de Web Modeling Language (WebML) y su aplicación a la construcción de WebRatio BPM, una herramienta basada en Eclipse que implementa una metodología para el desarrollo de

aplicaciones Web desde un diseño top-down, basada en procesos y dirigida por modelos.

### 4.3.3. Integración entre Procesos y Servicios

MINERVA (Model driven and Service oriented framework for the continuous business processes improvement & related tools) es una herramienta que soporta Business Process Service Oriented Methodology (BPMSOM) [109] definida como un plug-in para Eclipse. Este framework define transformaciones del tipo Query/ View/Transformation (QVT) entre modelos BPMN y modelos SoaML para generar servicios automáticamente [110].

En [1] se cuenta con el desarrollo de una metodología para generar un modelo para integrar aplicaciones dentro de una organización, alineando los procesos que definen sus operaciones con los servicios que soportan la funcionalidad. Basada en ocho etapas, es una metodología que enfatiza la identificación de requerimientos con enfoque de procesos. Cuenta con una etapa de modelado del negocio, previa al modelado de procesos y servicios. Luego de la identificación y modelado de servicios, define dos etapas más: diseño de componentes e implementación. Finalmente, cuenta con dos etapas globales: organización y planificación al comienzo y administración y monitoreo al final.

### 4.3.4. Discusión

Para concluir el análisis realizado, se describen los aspectos considerados en cada trabajo relacionado y se concluye en qué medida resultan incompletas en relación a esta propuesta o cómo han influido en la elaboración de la misma.

SOMA es un exhaustivo método altamente enfocado en la arquitectura orientada a servicios. En nuestra opinión, no considera en profundidad la especificación de requerimientos y la integración entre el ciclo de vida de los procesos y el ciclo de vida del software. SoaML se enfoca solamente en el modelado y diseño de los servicios de software, sin tener en cuenta el ciclo de vida de los procesos. Por lo tanto, comparado con nuestra propuesta, es una herramienta parcial. En relación a e3-value, nuestra propuesta sustenta un concepto similar. Sin embargo, nuestro enfoque enfatiza los procesos internos de una organización, usa notaciones estándar y adhiere al modelado top-down de los procesos de negocios combinado con un análisis bottom-up de los servicios. El proyecto Eclipse STP, por su parte, no propone la interacción entre los servicios y las actividades de los procesos. Las propuestas [105] y [106], además de ser propietarias de IBM, no incluyen la notación BPMN para modelar procesos, sino que transforman directamente las componentes ejecutables en lenguaje BPEL. WebRatio BPM se enfoca exclusivamente en el dominio de aplicaciones Web que, si bien son muy populares no son el único tipo de aplicaciones de software requeridas por muchas organizaciones. Más aún, la técnica top-down no contempla la posibilidad de que ciertas funcionalidades pre-existentes puedan ser descubiertas como parte de las actividades de los procesos de negocio. MINERVA es una solución completa y tiene un fuerte soporte para derivar



servicios pero está inmersa en un marco metodológico cuyos objetivos son diferentes a las propuestas en [1].

En conclusión, muchos de los trabajos relacionados existentes enfatizan los servicios. Nuestra propuesta pone en valor a los procesos de negocio como elementos que permiten seguir un ciclo integrado con los servicios mediante pasos de interacción. Estos pasos de interacción, combinan alternativamente enfoques top-down y bottom-up para modelar.

La Tabla 3 resume la discusión planteada en esta sección.

**Tabla 3. Comparación de Herramientas y Metodologías**

METODOLOGÍA/ HERRAMIENTA	CONSIDERA REQUERIMIENTOS	OBJETOS DE MODELADO	INCLUYE METODOLOGÍA	UTILIZA ESTÁNDARES
SOMA	No	Servicios	Si	No, es una metodología de IBM
SoaML	No	Enfoque MDD para servicios	Si	Si, define un perfil UML
e3-value	No	Modelado e interconexión de servicios	Si, a gran escala	No
ECLIPSE STP	No	Framework para el ciclo de vida de los servicios en SOA	No	Si
Rational Modeler IBM WebSphere	No	Transformación de procesos de negocios en SOA, expresados en UML	No	No. Exclusivo de IBM
WebRatio BPM	No	Metodología para desarrollo de aplicaciones web con perspectiva top-down, basada en procesos y dirigida por modelos	No	Sí, pero extiende BPMN
MINERVA	No	Servicios y procesos dentro de su propia metodología	Si	Si
MISP	Si	Ciclo de vida de procesos y servicios, de manera unificada integrando enfoques top-down y bottom-up	Si	Si

#### 4.4. MISP Y LAS MEJORAS INTRODUCIDAS

En esta sección, se describen las mejoras incorporadas a MISP en torno al modelado de procesos y servicios. Luego de una breve introducción, en la Sección 4.4.2 se describen los pasos de interacción de MISP. En la sección 4.4.3 se presenta el metamodelo P2S para conceptualizar servicios y en la sección 4.4.4 se presenta el modelado de componentes con SCA. Finalmente, en la sección 4.4.5 se muestra la integración entre BPMN, P2S y SCA.

##### 4.4.1. Introducción

Como se describe en [1], la etapa de modelado del negocio produce un mapa de procesos que, junto con los requerimientos identificados y especificados y los casos de uso del negocio, forma la base para el modelado de procesos del negocio.

A partir del modelo del negocio, el modelado de procesos de negocio es un paso de refinamiento, donde los procesos son enriquecidos con la definición de aspectos funcionales del sistema mediante casos de uso y la descripción de los objetos de información. Este paso de refinamiento permite dividir los procesos en tareas y definir roles y documentación textual que contribuyen al BPD. Una vez obtenido el modelo de procesos, se encuentra el terreno propicio para identificar los elementos funcionales (servicios) que serán consumidos por los procesos de negocios. La Figura 15 muestra las 8 etapas definidas en [1] y resumidas en la Sección 4.2, ubicadas de manera superpuesta sobre las fases del ciclo de vida del software. La Figura 16 representa cómo interactúan las etapas entre sí, ambos desde el ciclo de vida del software (columnas).

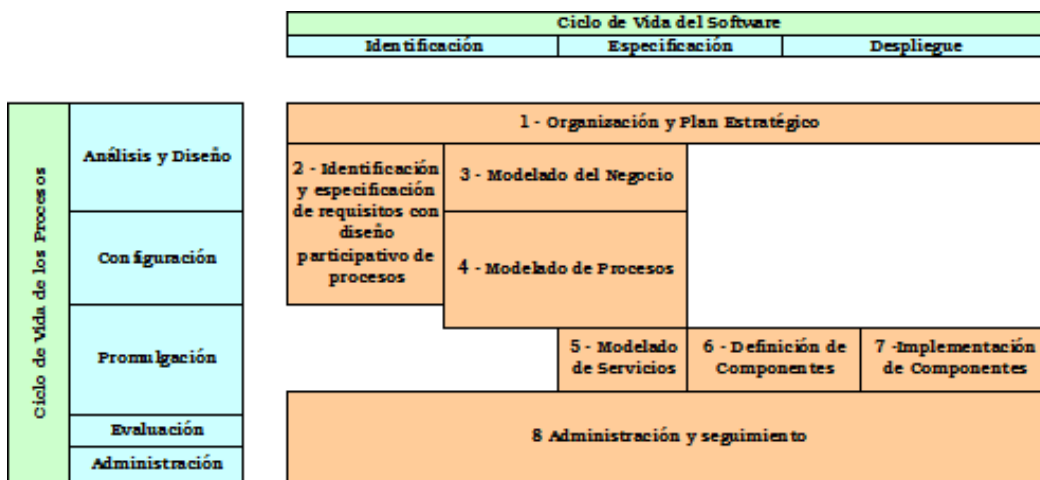


Figura 13. Marco Metodológico Propuesto: Etapas en Cada Ciclo de Vida

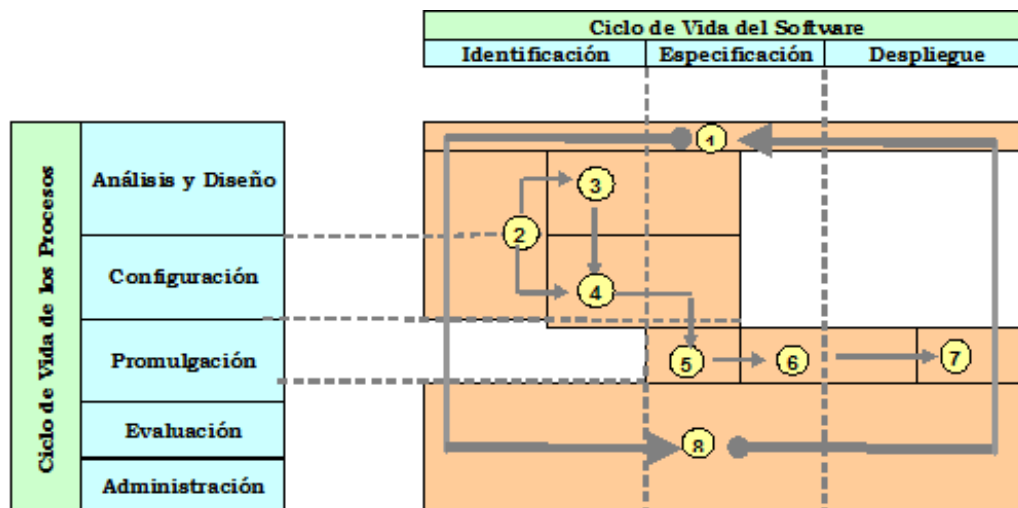


Figura 14. Marco Metodológico Propuesto: Interacción de Etapas

**Modelado de proceso y clasificación**

El corazón del modelado de los procesos de negocio son los procesos en sí mismos. Aplicando diferentes niveles de abstracción, existen tres conceptos relacionados: 1) procesos, 2) sub-procesos y 3) tareas. Un proceso es una red de “cosas para hacer” o actividades. Un sub-proceso es un proceso en sí mismo cuya funcionalidad es parte de un proceso más grande. Los procesos de menor nivel que no pueden ser descompuestos, son las tareas o actividades.

El metamodelo de BPMN presentado en la Figura 17 describe las actividades de los procesos como generalización de subproceso y actividades.

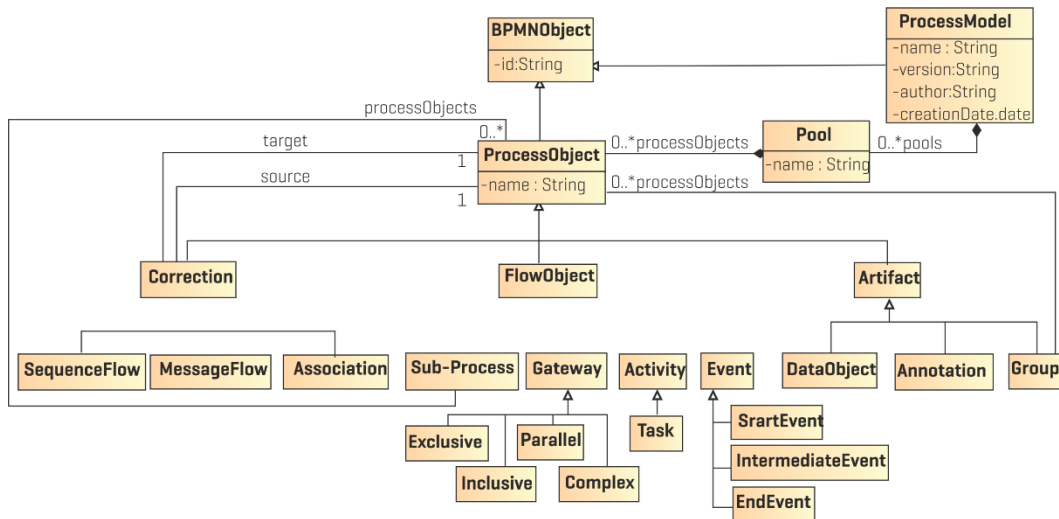


Figura 15. Metamodelo Simplificado de BPMN

### Modelado y clasificación de servicios

Desde el punto de vista del paradigma de objetos, los servicios son objetos que proveen más información y debido a su bajo acoplamiento, generalmente no admiten asociaciones, pero cuyo comportamiento es disparado por su meta-información, básicamente en términos de contratos de servicios y contextos de ejecución. Sin embargo, su similitud con los objetos hace posible modelar en términos de sus interfaces.

Como se mencionó anteriormente, Erl define los símbolos de servicios como un círculo comprendido por dos áreas: un área para el nombre del servicio y otra para sus capacidades [100].

Los contratos de servicio indican las interfaces que proveen, sus operaciones y parámetros, favoreciendo la catalogación de los servicios y utilizando la composición para extender la funcionalidad.

La granularidad de un servicio comunica el nivel de detalle del mismo asociado a su alcance funcional. No existe un único criterio que indique el nivel de granularidad adecuado para un servicio, pero los principios de diseño y las clasificaciones de servicios que se adopten impactan en forma directa en el nivel de granularidad.

La clasificación propuesta en nuestro metamodelo tiene en cuenta servicios de: negocio, entidad y de infraestructura:

- *Servicios de negocio* - tienen como alcance funcional las entidades del negocio, son de alta reusabilidad y son agnósticos a muchos procesos de negocios.
- *Servicios de entidad* - tienen límites funcionales asociados a las clases dentro de un modelo de clases. Tienen menos reuso potencial y se componen con otros servicios.
- *Servicios de infraestructura* - son servicios que resuelven servicios de tecnología o aspectos transversales. A modo de ejemplo, un servicio que establece la conexión con una base de datos relacional en forma genérica o un servicio centralizado que informa condiciones de excepción, ambos son ejemplos de servicios transversales.

Los servicios son agrupados en función de un comportamiento común, en vez de encapsular dicho comportamiento junto con los datos, como sucede en la orientación a objetos.

#### **4.4.2. Pasos de interacción**

Como se menciona en la Sección 4.1, una vez modelados los procesos como un conjunto ordenado de actividades, los servicios son modelados para proveer la funcionalidad requerida por tales procesos. El modelado de servicios comienza con la identificación de los mismos.

La identificación de servicios puede abordarse con un enfoque bottom-up. La idea de este enfoque es considerar a los servicios como elementos de alcance funcional bien definido y que se acceden uniforme y completamente independientemente de donde son utilizados [92]. El enfoque bottom-up permite tener en cuenta componentes funcionales individuales y su composición en una estructura global.

Por su parte, los procesos de negocio son modelados en etapas tempranas siguiendo un enfoque top-down. Este enfoque comienza con elementos más generales (procesos) y los descompone en componentes con un alto grado de refinamiento (sub-procesos y tareas) logrando una perspectiva global y abarcativa.

Por estas razones concluimos que el modelado de procesos adhiere a un enfoque top-down y el modelado de servicios tiene características bottom-up. Por lo tanto, estos pasos de interacción para el modelado de procesos y servicios muestra la necesidad de contar con un mecanismo que ayude a mantener este ciclo iterativo top-down / bottom-up. Para proveer este mecanismo, se propone la definición de un metamodelo para conceptualizar servicios. Este metamodelo facilita la interacción entre servicios y procesos modelados en BPMN.

#### 4.4.3. Metamodelo de Conceptualización de Servicios

El metamodelo propuesto se basa en una adaptación del metamodelo de conceptualización de servicios Service Oriented Architecture Foundation (SOAF) [112] (Figura 18) y el metamodelo de BPMN [113] (Figura 19). El modelo propuesto se basa en la consideración de los procesos como un conjunto de actividades donde cada actividad es una abstracción de funcionalidad que será “realizada” por un servicio. El metamodelo es denominado ProcessToService (P2S) y se esboza en la Figura 19. El mismo, conceptualiza servicios como una generalización de componentes internas y servicios externos. Se construye como una instancia del estándar MOF (Meta Object Facility) [96], que constituye el nivel más abstracto de la arquitectura de modelado de 4 capas definidas por OMG [39].

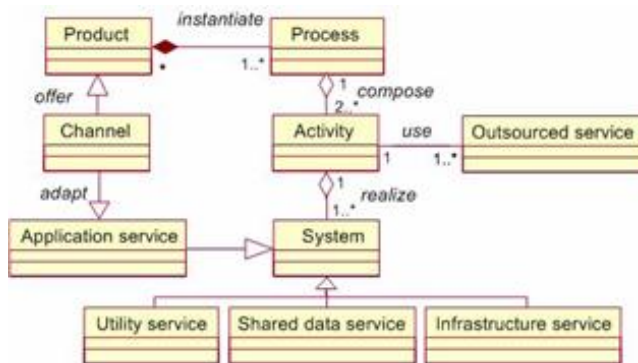


Figura 16. Metamodelo de Conceptualización de Servicios SOAF

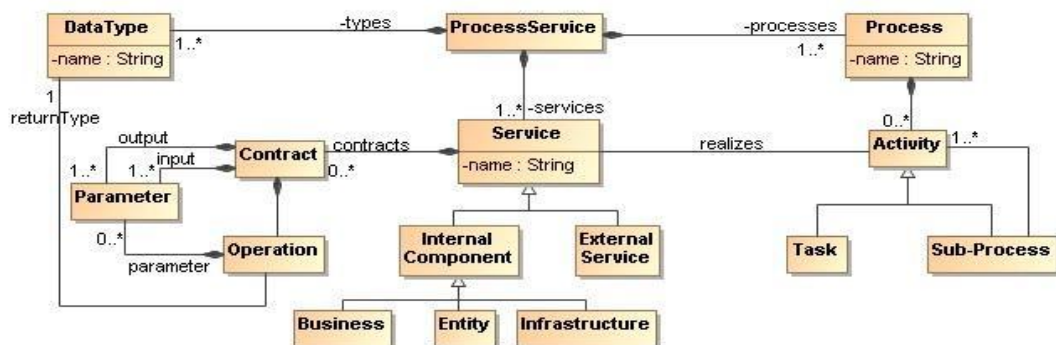


Figura 17. P2S Metamodelo de Conceptualización de servicios

El proceso de obtener servicios desde las actividades de un proceso de negocios es un punto clave en la interacción entre procesos y servicios. El metamodelo P2S determina que la metaclase Services “realiza” una actividad. Por lo tanto, puede aplicarse la combinación top-down y bottom-up descrita en la sección anterior. Esto es, después de aplicar un análisis top-down para identificar actividades, se aplica un análisis bottom-up para obtener las

componentes funcionales que responden a las actividades de negocio. La premisa para determinar una componente funcional es que contiene la información a ser administrada, las operaciones requeridas para manipular tal información y las reglas de negocio que gobiernan tal manipulación. Después este análisis bottom-up, determinará qué actividades serán realizadas por qué servicios. Así, cada servicio es clasificado según la generalización de metaclases en el metamodelo. Los servicios tienen un nombre y una colección de contratos, que, a su vez tienen un conjunto de interfaces que implementan las componentes. La Figura 19 muestra que los servicios son realizados por subprocesos o tareas.

#### 4.4.4. Definición de componentes de MIS P

En el marco de MIS P [1], los servicios deben encapsular la lógica dentro de un contexto que puede ser una tarea, una entidad de negocio o algún otro agrupamiento a fin de mantener su independencia. Para que los servicios puedan orquestarse y desplegar su lógica de funcionamiento, deben intervenir en la ejecución de las actividades del negocio, para lo cual deben poder establecerse relaciones con aquellos que quieren usarlos.

Concluida la etapa de identificación y modelado de servicios, continúa la etapa de definición de componentes. En esta etapa, se empaquetan los servicios como componentes de granularidad gruesa, para dar respuesta a los requisitos funcionales y no funcionales identificados en etapas iniciales y que son utilizados por procesos de negocios que los requieren.

#### SCA y su metamodelo

SCA define un enfoque general para crear componentes y describir cómo éstos trabajan juntos [2.25]. Siendo actualmente un estándar de OASIS [115], fue creado originalmente por un grupo de representantes de la industria, incluyendo, entre otros, BEA, IBM, Oracle y SAP.

Las especificaciones SCA definen cómo crear componentes y cómo combinarlas en aplicaciones completas. Los componentes pueden ser construidos con JAVA y otros lenguajes usando los modelos de programación basados en SCA, o pueden construirse con otras tecnologías, como BPEL (Business Process Execution Language) o el framework Spring. Más allá de la tecnología de componentes usadas, SCA define un mecanismo común para especificar cómo se combinan componentes en aplicaciones de software.

En la Figura 20 se presenta un metamodelo simplificado del estándar SCA definido en [114]. Allí se observa que: un *Composite* contiene *Reference*, *Component*, *Wire*, *Service* y *Property*. A su vez un *Component* contiene *ComponentReference* y *ComponentService*. *Reference* y *ComponentReference* son subclases de *BaseReference* y *ComponentService* y *Service* son subclases de *BaseService*, que contiene *Operation*. *Reference* y *Wire* se relacionan con *ComponentReference*. A su vez, *Service* y *Wire* se relacionan con

*ComponentService*. *Component* contiene *PropertyValue*, mientras que *ComponentType* contiene *Property*.

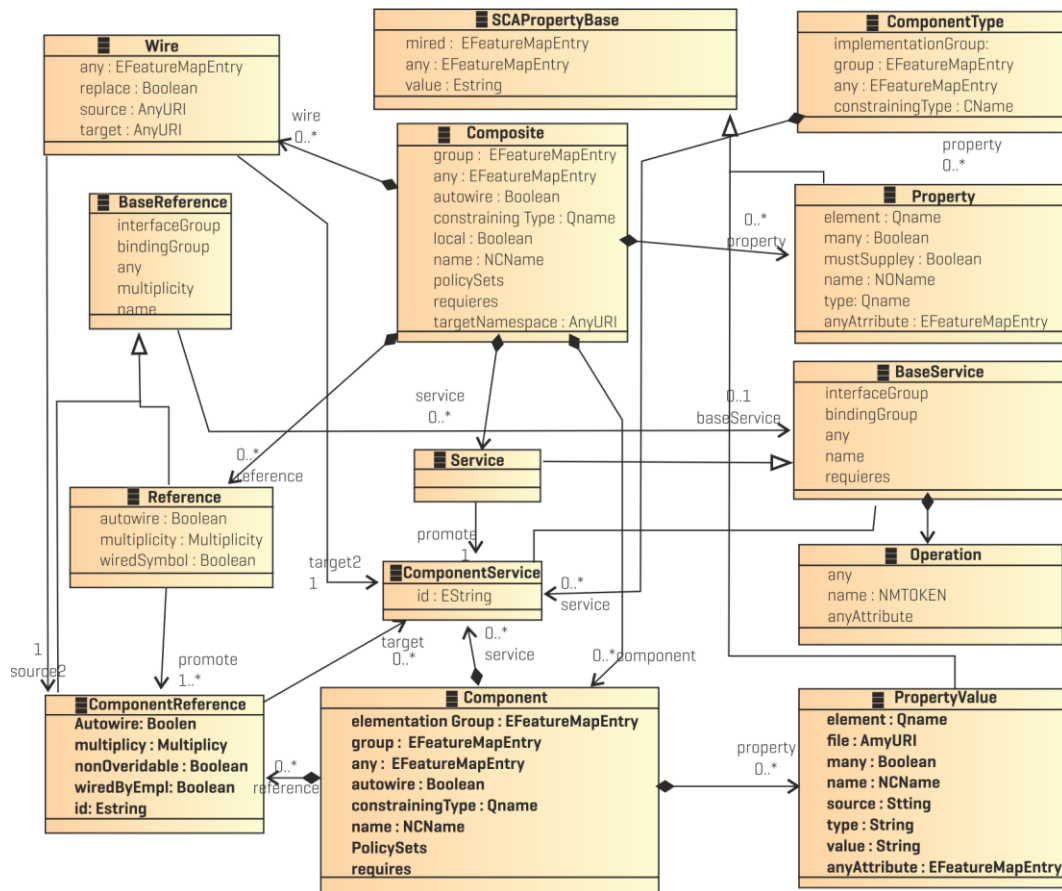


Figura 18. Metamodelo Simplificado de SCA

## Descripción y representación gráfica de los elementos

Dentro del estándar SCA, los elementos metamodelados poseen una representación gráfica y una descripción que se detalla a continuación.

### Compuestos

Un compuesto es una estructura lógica, donde sus componentes pueden ejecutarse dentro de un único proceso y en un único computador, o en procesos y computadores diferentes. Una aplicación puede definirse como un único compuesto o combinando varios compuestos. Además los componentes que se usan en cada compuesto pueden compartir tecnologías o ser de tecnologías diferentes. Un compuesto SCA se describe con un archivo XML de configuración, con extensión *.composite* y que se denomina Service Component Definition Language (SCDL), pronunciado “skiddle”.

### Componentes

Los componentes son las unidades atómicas desde las cuales se construye una aplicación SCA. Un componente es una instancia de una implementación que ha sido configurada adecuadamente. La implementación es el código que realmente provee de funcionalidad al componente - como un clase Java o un proceso BPEL. La configuración, expresada en SCDL, define cómo el componente interactúa con el mundo externo. Cada componente se sustenta sobre un conjunto de abstracciones que incluyen servicios, referencias, propiedades y enlaces para especificar su interacción con el exterior (como se observa en la Figura 21). A su vez, cada componente implementa una lógica de negocio expuesta como servicio (representado por el símbolo verde). Cada servicio provee una cantidad de operaciones que pueden ser accedidas por el cliente del componente. La manera de describir el servicio depende de la tecnología para implementar el componente. Por ejemplo, si el componente es una clase Java, los servicios se describen usando interfaces, mientras que si se trata de un componente implementado en BPEL, se describe usando WSDL.

Además de proveer servicios a sus clientes, los componentes pueden consumir servicios de otros componentes. Esto se describe indicando los servicios mediante referencias (símbolos violetas). Cada referencia define una interface que contiene operaciones que el componente necesita invocar.

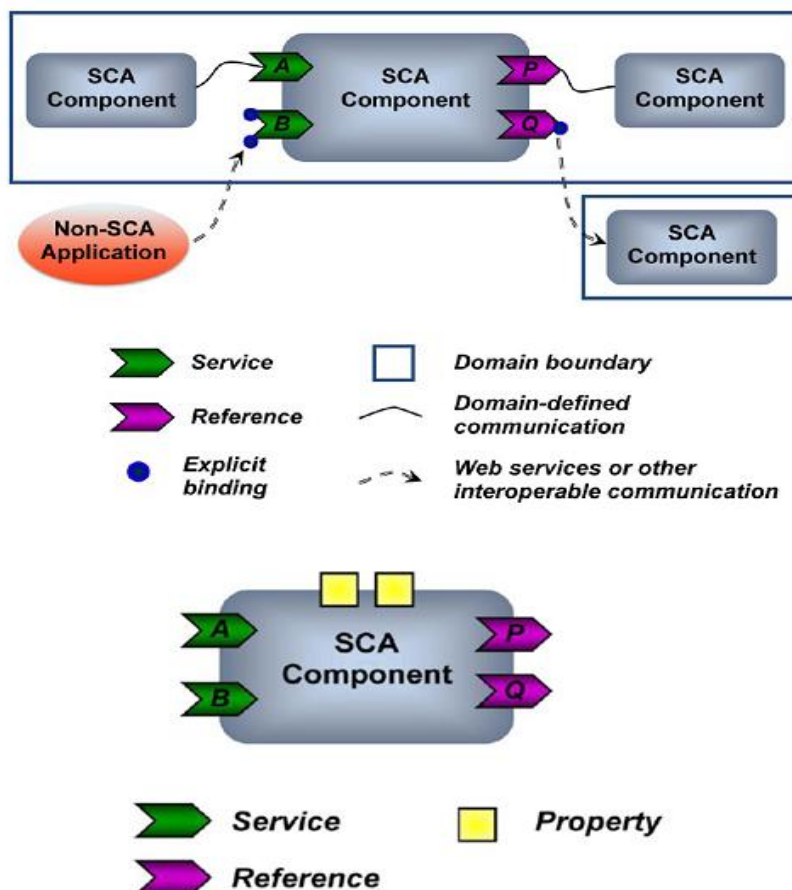


Figura 19. Enlaces y Bindings SCA [114]



Dado que se trata de un metamodelo simplificado, es posible que algunos elementos no estén completos, pero esta simplificación resulta suficiente para analizar la integración con el metamodelo de conceptualización de servicios (P2S).

Uno de los aspectos más evidentes al analizar una integración entre P2S y SCA proviene del hecho que *Component* en P2S es una especialización de *Service*, mientras que en SCA, tanto *Component* como *Service* son parte de *Composite*. Es decir, *Service* no comparte el mismo nivel de abstracción en los dos metamodelos.

Más allá de este punto, se observa que el estándar SCA permite, a través de su metamodelo, definir un lenguaje de especificación para la etapa 6 de la metodología (Definición de componentes). La interacción entre esta etapa y la anterior (etapa 5 – Modelado de Servicios) se podrá resolver instanciando ambos metamodelos, y poniendo cuidado para resolver la ambigüedad del concepto *Service*.

La solución propuesta para resolver la ambigüedad de este elemento es definir que *Service* en P2S representa un modelo de servicio y no el servicio en sí mismo. Se trata de un servicio conceptual que puede ser un componente o un servicio externo (tal como se describe en P2S). En este sentido, la metaclass *Service* de P2S pasa a nombrarse como *ConceptualService* y así se puede vincular ambos metamodelos a través de la metaclass *Component*.

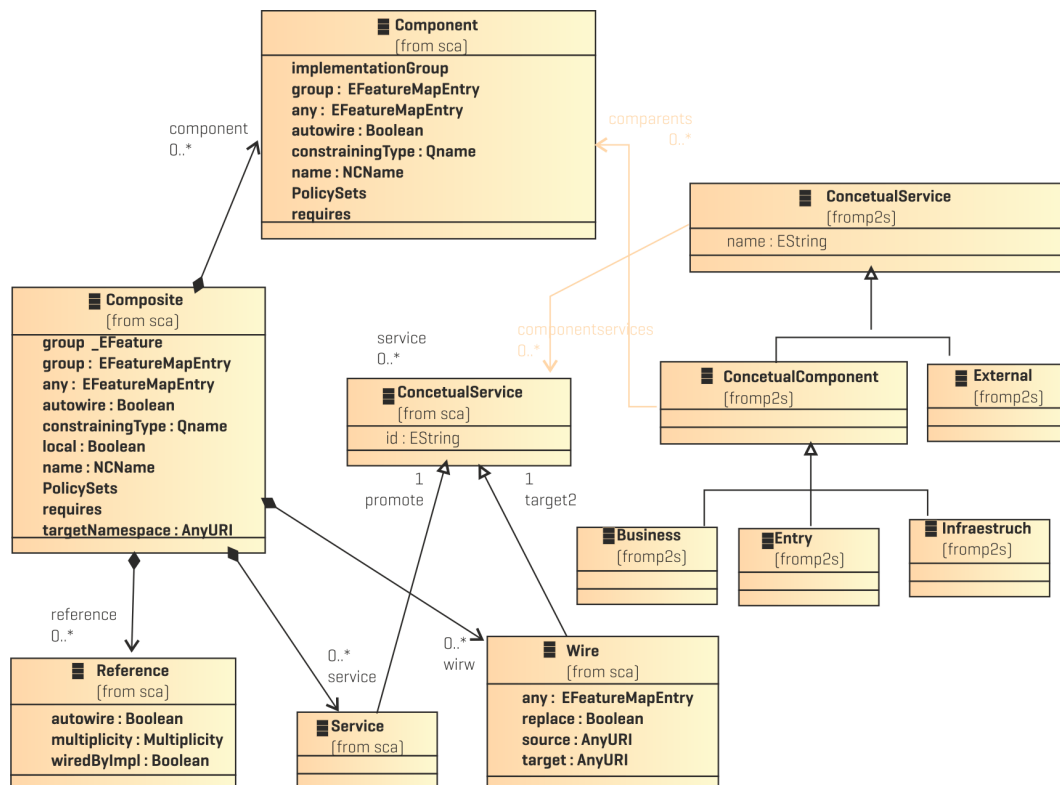


Figura 20. Vinculación del Metamodelo P2S con SCA

A su vez, la metaclassa *Component* que se utiliza como medio de vinculación entre los dos metamodelos, puede también renombrarse como *ConceptualComponent* en P2S y así se puede relacionar con *Component* en SCA. De la misma manera, se define *ConceptualService* en P2S para relacionarlo con *ComponentService* en SCA. La Figura 20 presenta una versión simplificada de esta propuesta de integración de metamodelos.

A la luz de lo expuesto, la Tabla 4 presenta un conjunto de reglas de vinculación entre las metaclasses de cada metamodelo. Estos mapeos constituyen un primer paso en la definición de transformaciones entre metamodelos. Si bien el desarrollo de este punto será objeto de investigaciones futuras, presentamos a continuación una breve explicación de cada mapeo.

**Tabla 4. Reglas de Integración entre Procesos y Servicios**

METACLASES BPM	METACLASES P2S	METACLASES SCA
Process Model	Process	Composite
Pool	ProcessService	Component
Connection	Contrato	Wire
Artifact	TypedElement	Reference
Activity	ConceptualService	ComponentService

Como mencionamos en la Sección 4.4.4, un modelo SCA consiste de una serie de compuestos que definen la configuración de un dominio. Este dominio representa las capacidades o funcionalidades del negocio que son controladas por una organización y son el insumo para ejecutar las actividades de un proceso de negocio. En este sentido, un Compuesto SCA equivale a un Proceso en el metamodelo P2S y es a su vez el *ProcessModel* de BPMN. Este mapeo conceptual da cuenta que en realidad cada proceso de negocio tiene su representación de implementación como un Compuesto SCA.

El mapeo de Pool a *Component* se sustenta en la idea de que *Component* es un metaclassa contenedora de *ComponentService* y por lo tanto, puede desagregarse en más partes. Además, Pool puede interpretarse como un participante de un proceso de negocio que produce una función de negocio esperada; siendo ésta última un *Component*.

Las actividades de los procesos (*Activity*) se mapean a *ComponentService* que son la implementación de un servicio como instancia para proveer o consumir funciones básicas de negocios. Por último, *Connection* y *Artifact*, se mapean a *Wire* y *Reference* como representación conceptual de los flujos que permiten orquestar el proceso.

#### 4.4.6. De los Componentes a los Servicios Web

Para formalizar implementación de componentes, es posible utilizar Web Service Definition Language (WSDL) [120]. La tecnología de servicios web es una alternativa para componer y desplegar servicios estándar. Los documentos

WSDL describen la funcionalidad del servicio web y contiene información sobre la localización del servicio, métodos y protocolos.

La nueva versión, WSDL 2.0 provee un modelo y un archivo XML para describir los servicios web. WSDL 2.0 permite separar la descripción de la funcionalidad abstracta ofrecida por el servicio, de los detalles completos del mismo tal como “como” o “donde” la funcionalidad se ofrece. En [121] se define un lenguaje para describir la funcionalidad abstracta de un servicio así como un framework para describir los detalles completos del servicio descrito. También define los criterios conformidad para los documentos en este lenguaje.

Analizamos el metamodelo de WSDL en torno a las metaclases que ayudan a integrarlo con el metamodelo de SCA y así obtener la definición de los servicios Web que implementen las componentes.

WSDL 2.0 describe un servicio Web en dos etapas: una abstracta y una concreta. A nivel abstracto, WSDL 2.0 describe un servicio Web en términos de los mensajes que envía y recibe; los mensajes son descritos independientemente del formato usando un sistema de tipos, típicamente basado en XML. Una *operation* asocia un patrón de mensajes de intercambio con uno o más mensajes. Una *interface* agrupa operaciones sin formato de transporte. A nivel concreto, un *binding* especifica el formato de transporte para una o más interfaces. Un *endpoint* se asocia a una red dirigida por el *binding*. Finalmente, un *service* agrupa los *endpoints* e implementa una interface común.

El modelo conceptual WSDL es muy simple: hay una definición de componentes que contiene componentes WSDL 2.0 (*interfaces, bindings and services*) y las componentes de tipos del sistema (*element declaration and type definitions*). La Figura 23 representa lo detallado anteriormente.

Integramos el metamodelo SCA con el metamodelo WSDL a través de las metaclases *Definition, Service* y *Operation* de WSDL. La metaclase *Component* de SCA, se vincula con cada *Definition* en WSDL. Cada *ComponentService* que pertenece a *Component* en SCA se vincula con cada *Service* en WSDL. Finalmente, cada *Operation* en SCA, mapea a una *Operation* en WSDL.

El lenguaje WSDL tiene un metamodelo [122] [123] que está integrado a otros. En consecuencia, provee un mecanismo para formalizar la Etapa de Implementación de Componentes de MISP. La Figura 24 muestra la integración propuesta entre los metamodelos P2S, SCA y WSDL.

Si bien es cierto que WSDL es simplemente un lenguaje de descripción de Servicios Web y que éstos últimos son solamente una manera de implementar servicios en una arquitectura SOA, nuestra propuesta se sustenta en el hecho de que al proponer un enfoque metodológico integrador que cuenta con una etapa de modelado de servicios con P2S, es en esta instancia en que pueden descubrirse la existencia de piezas de software existentes para ser consumidas por las actividades de un proceso. Este activo de software puede ser encapsulado fácilmente como Servicio Web, aunque no haya sido concebido como componente SCA.

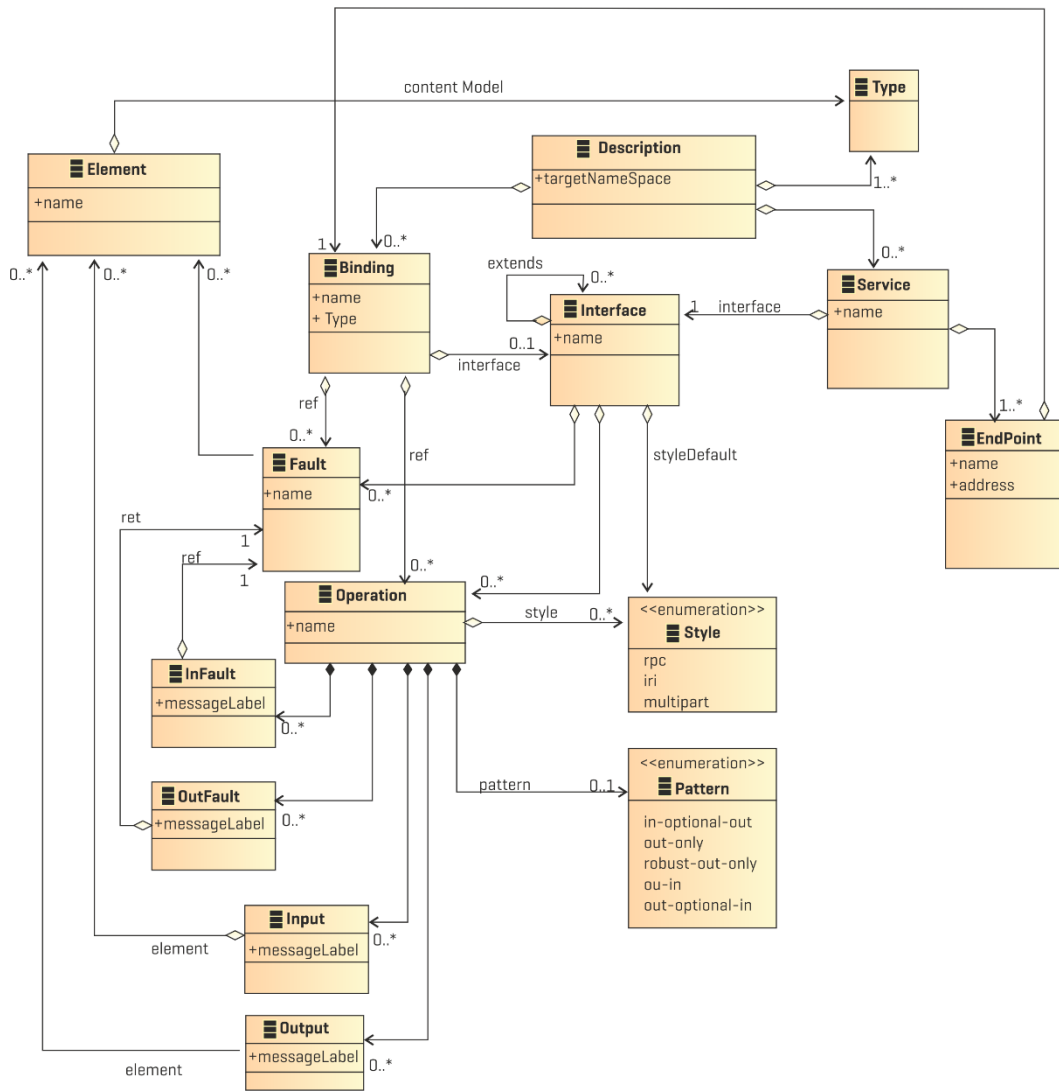


Figura 21. Metamodelo WSDL 2.0

De este modo, la propuesta de obtener una formalización de la metodología con vistas a producir transformaciones entre metamodelos, absorbe a nuestro juicio buena parte de los activos de software existentes en las organizaciones de hoy, incluso en aquellos casos en que el paradigma de solución no sea estrictamente orientado a servicios.

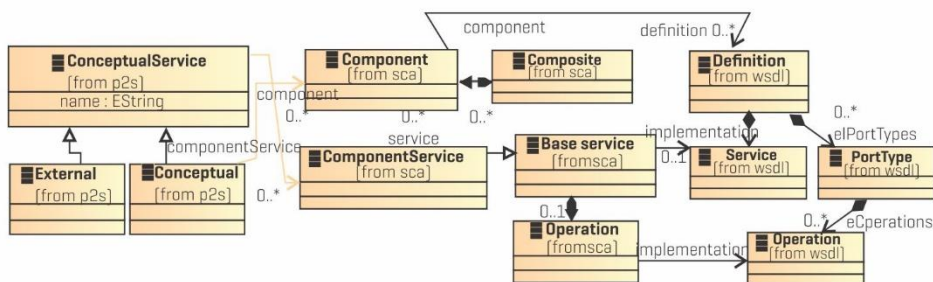


Figura 22. Integración entre los Metamodelos P2S, SCA y WSDL

## **4.5. MISP Y SU MEJORA A TRAVÉS DE METAMODELOS. UN EJEMPLO**

En esta sección presentamos un ejemplo donde se aplican las mejoras introducidas a la MISP. En el mismo, se observa la ventaja de contar con un lenguaje para describir servicios que integre las actividades del proceso con las componentes de software que lo implementan. Esta integración permite, además, conceptualizar de manera bottom-up a partir de un análisis orientado a objetos, las clases subyacentes al proceso que carecen de representación en BPMN debido a que es una notación orientada a la representación de actividades y flujos y no a la estructura de los datos ni a su funcionalidad.

El ejemplo modela un proceso que se desarrolla en el centro de reparaciones de una compañía de transporte, cuando un vehículo requiere reparación a través de su chofer. Se presenta el modelado en BPMN, se identifican sus actividades y a partir de allí, se determinan los servicios que “realizan” dichas actividades modelando y clasificando servicios con P2S. Luego se modelan las componentes utilizando la integración entre P2S y SCA.

### **4.5.1. Descripción del Problema**

En un centro de reparaciones de una empresa de transportes, se llevan a cabo una serie de actividades vinculadas la reparación de los vehículos de dicha empresa que se inicia cuando cada conductor se acerca al centro de reparaciones (taller) para solicitar una reparación.

Esta solicitud genera una orden de trabajo (OT) que identifica a dicha solicitud. El responsable del taller determina los ítems de la OT según el desperfecto identificado y detalla los materiales requeridos. Luego de haber generado la OT, se determina si puede planificarse dicha reparación en función de la disponibilidad de materiales y de los mecánicos idóneos según el desperfecto. Si esta planificación es posible, se programa la asignación de un box y de un mecánico por cada ítem de la OT. Luego, se realiza la reparación calculando costo real y material utilizado y luego se cierra la OT, dando por terminada la solicitud.

### **4.5.2. Modelado del Proceso en BPMN**

El modelado del proceso implica determinar quiénes están involucrados en el mismo y qué actividades realiza cada uno de ellos. A partir de allí se analiza cuáles actividades son tareas y simples y cuales, por su complejidad, ameritan modelarse como un subproceso. Con estos insumos, se describe en notación BPMN el diagrama del proceso donde se incluyen las condiciones bajo las cuales se bifurcan los flujos entre las actividades. Este análisis representa la actividad clásica de modelado de procesos y es una tarea que no requiere conocimientos técnicos. Solo es necesario conocer los elementos que

constituyen la notación BPMN y su semántica. A partir de estas premisas entonces se identifican los participantes del proceso que son el chofer y el taller.

Las actividades a realizar se pueden enumerar como:

Generar OT, que es la primera actividad que se ejecuta ante la llegada de una solicitud de reparación. Esta actividad implica una revisión por parte del mecánico que no requiere ser reflejada dentro del proceso, solo se deja registro de los ítems que se asocian a la OT según el resultado de dicha revisión. En este sentido, se interpreta como una tarea simple. El participante involucrado es el Chofer.

Planificar reparación, involucra un conjunto de actividades más detalladas y que considera la interacción con otros, disponibilidad de materiales, cálculo de costo de cada uno de ellos y especialidad del técnico involucrado en cada ítem de la OT. Dada esta complejidad, se modela esta actividad como un sub-proceso. Además, aparte de lo complejo de la misma, el modelado como sub-proceso permite detallar con mayor precisión las tareas permitiendo generar rastros detallados de su ejecución que contribuyen a producir indicadores de rendimiento útiles para la empresa. El participante involucrado es el Taller.

Asignar mecánico, es una tarea simple que determina quien está a cargo de cada ítem de la OT y que resulta de una planificación exitosa. El participante involucrado es el Taller.

Efectuar la Reparación, es otra tarea simple que solo registra en el procesos, el estado de cada ítem de la solicitud. El participante involucrado es el Taller.

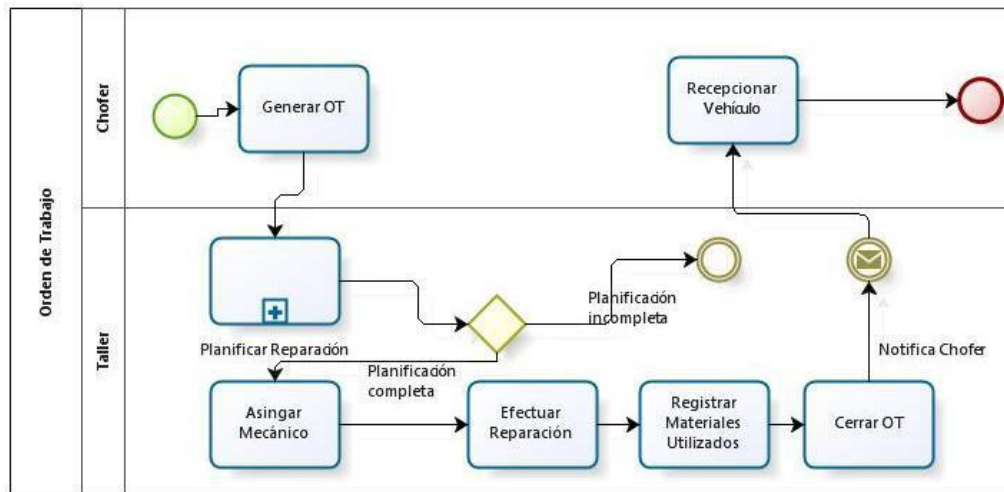
Registrar los Materiales Utilizados, es una tarea simple que indica, para cada ítem, la cantidad real de materiales utilizados. El participante involucrado es el Taller.

Cerrar la Orden de Trabajo, esta también una tarea simple que registra la concreción de cada ítem de la OT y en consecuencia el estado final de la misma, calculando el costo real de la reparación. Cabe aclarar que si bien podría haberse considerado una actividad lo suficientemente compleja como para modelarse como sub-proceso, se resuelve que el procedimiento interno no aporta rastros de interés para los indicadores de calidad que podrían considerarse para el proceso, debido a que solo se realizan cálculos aritméticos entre los datos que se registran. El participante involucrado es el Taller.

Recepcionar Vehículo, es una tarea simple que deja registro que el vehículo fue retirado del taller y se cierra definitivamente la solicitud iniciada. El participante involucrado es el Chofer.

En la Figura 26 se observa el modelo en BPMN del proceso enunciado. En dicho modelo se incluye la bifurcación que se produce como resultado de una planificación incompleta, generando y evento Espera, o bien prosiguiendo con la ejecución del proceso a la próxima actividad. También se modela un evento de

Notificación, que dispara la ejecución de la actividad *Recepcionar Vehículo* al *Chofer* correspondiente, según la solicitud que guía el flujo de trabajo.



**Figura 23. Proceso de una Orden de Trabajo**

#### 4.5.3. Modelado y Conceptualización de Servicios

El proceso de obtener servicios a partir de las actividades de un proceso de negocio es uno de los puntos clave en la interacción entre procesos y servicios. El metamodelo P2S determina que la metaclassa Servicio “realiza” una Actividad.

Este aspecto se pone en práctica aplicando una combinación top-down y bottom-up. Es decir, una vez aplicado el análisis top-down para identificar las actividades, se aplica un análisis bottom-up para obtener los componentes funcionales que dan respuesta a las actividades del negocio.

Este análisis bottom-up parte de revisar, para cada actividad, cuales son las componentes funcionales que se requieren y cuál sería su comportamiento. Se construye desde abajo, dado que se debe determinar cuáles componentes existen y cuales deben producirse como funcionalidades nuevas, para luego clasificarlas según su comportamiento siguiendo el criterio definido en la Sección 4.4.3.

En el caso de componentes nuevas se puede partir de la identificación de clases y sus métodos siguiendo un análisis orientado a objetos, dada las similitudes entre el diseño orientado a objetos y el orientado a servicios tal como lo expresa Erl en [100] y teniendo en cuenta las restricciones que el mismo autor plantea a la hora de conceptualizar las clases como servicios.

En este sentido, emergen del análisis, dos servicios que representan la estructura de datos subyacente en el proceso y que guía buena parte del flujo del mismo y que pueden conceptualizarse como clases o servicios de entidad (según nuestra clasificación conceptual de los servicios). Estos son el Servicio

Orden de Trabajo administra las OT y sus ítems de manera unificada, tal como ocurre con el Servicio Conductor.

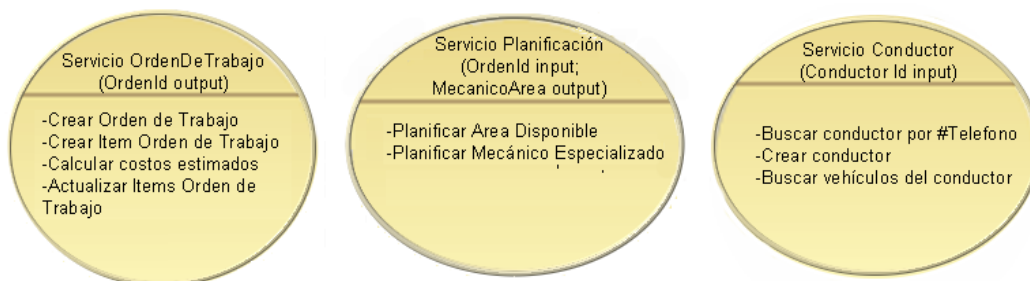
Por otro lado, el sub-proceso Planificar Reparación, indica un grado de complejidad que incluye varias actividades propias del dominio del problema o negocio, e incluso, como se mencionó, puede involucrar la definición de indicadores de rendimiento que aporten a la mejora de procesos. En este sentido, se trata de un servicio de negocio, tal como lo define P2S y se nombra como Servicio Planificación.

En la Figura 27 se modelan los servicios descritos, indicando su interface y sus capacidades. La premisa es que los componentes contengan la información a ser gestionada: 1) las operaciones para manipular la información, y 2) las reglas de negocio que gobiernan dicha manipulación.

Luego de este análisis bottom-up, se determinan cuáles actividades son realizadas por cada servicio identificado, el cual, a su vez, estará clasificado según la generalización de metaclases del metamodelo.

En la Figura 27, los servicios son identificados agrupándolos por comportamiento, siguiendo el criterio definido en la Sección 4.4.3. Esta construcción conceptual sigue un enfoque bottom-up por este agrupamiento de comportamiento aparece desde las clases identificadas mediante un análisis orientado a objetos. Por lo tanto el Servicio Orden de Trabajo administra las OT y sus ítems de manera unificada, tal como ocurre con el Servicio Conductor. Ambos son servicios de entidad. Además, el Servicio Planificación será el responsable de planificar las órdenes, un servicio de negocio.

Siguiendo con el análisis orientado a objetos, los servicios de entidad, como Orden de Trabajo y Conductor, aparecen desde las clases que modelan las entidades del negocio. Este enfoque permite aislar los aspectos funcionales que se mantienen ligados a las actividades del proceso de negocios.



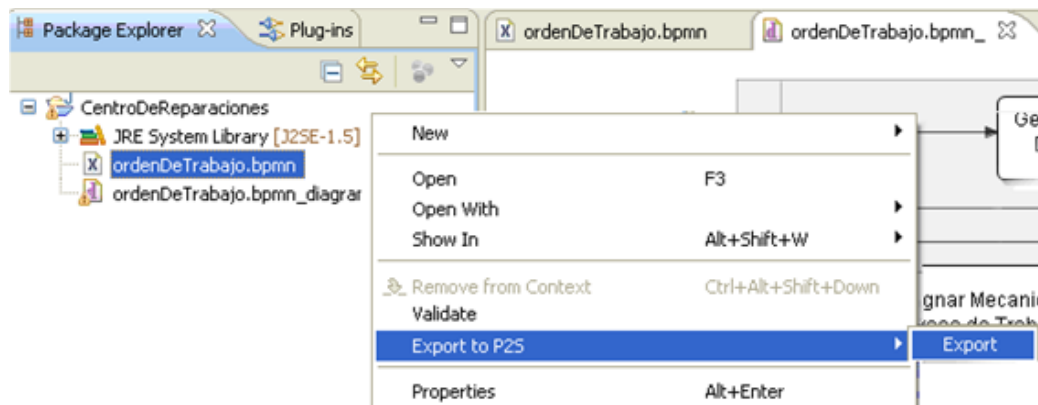
**Figura 24. Servicios del Proceso de OT**

Esta fase de conceptualización de servicios se modela a través de P2S que representan un lenguaje propio para clasificar servicios y vincularlos a las



actividades de los procesos. Para implementar este metamodelo se construye un prototipo de editor gráfico de servicios para lo que se utilizó un plugin de Eclipse incluido en el proyecto Eclipse Modeling [13][12][119]. Este prototipo permite definir los servicios arrastrando y soltando elementos de una paleta, donde se distinguen con distintos colores cada uno de los tipos de servicios (de entidad, de negocio, etc.), permite describir sus parámetros y capacidades y vincularlos con las actividades del proceso correspondiente definiendo la relación “realiza”

A partir del modelo de procesos definido en la sección anterior que fue implementado en un editor gráfico basado en Eclipse, se encuentra que dicho editor divide la información en dos archivos: uno de ellos contiene las instancias de los elementos del proceso (con extensión BPMN) y el otro contiene la información gráfica. En la Figura 28 se muestra como la información requerida para el modelado de servicios puede exportarse desde el archivo bpmn a P2S. La información se exporta en dos archivos, siguiendo los mismos criterios utilizados para BPMN. Luego el prototipo utiliza tales archivos para editar gráficamente los servicios.



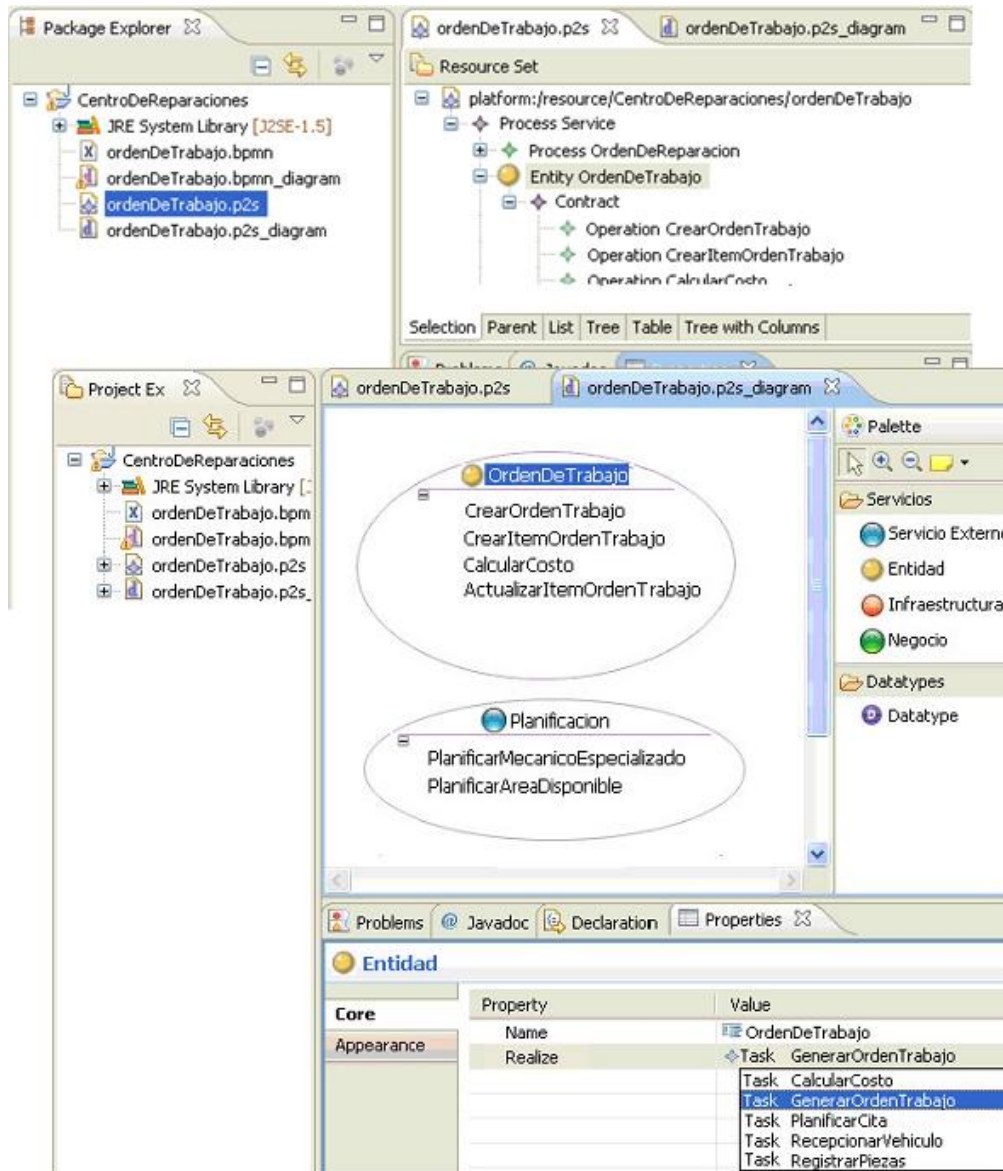
**Figura 25. Exportando la OT desde BPMN a P2S**

En la Figura 29 se muestra el ejemplo como una instancia del metamodelo definido en la Figura 17. Esta instancia contiene algunas de las actividades identificadas en el modelo de proceso de la Figura 23 y de los servicios modelados en la Figura 24. La misma figura muestra el editor gráfico con algunas instancias del ejemplo. También resalta la instancia del servicio de entidad llamado Orden de Trabajo. Como indica la aplicación del método, el prototipo permite asociar servicios a actividades de procesos mediante la propiedad “realiza”.

#### 4.5.4. Definición de Componentes

En el marco de MISP, a fin de mantener su independencia, los servicios deben encapsular la lógica dentro de un contexto que puede ser una tarea, una entidad de negocio o algún otro agrupamiento. Para que los servicios puedan orquestarse y desplegar su lógica de funcionamiento, deben intervenir en la ejecución de las actividades del negocio, para lo cual deben poder establecerse

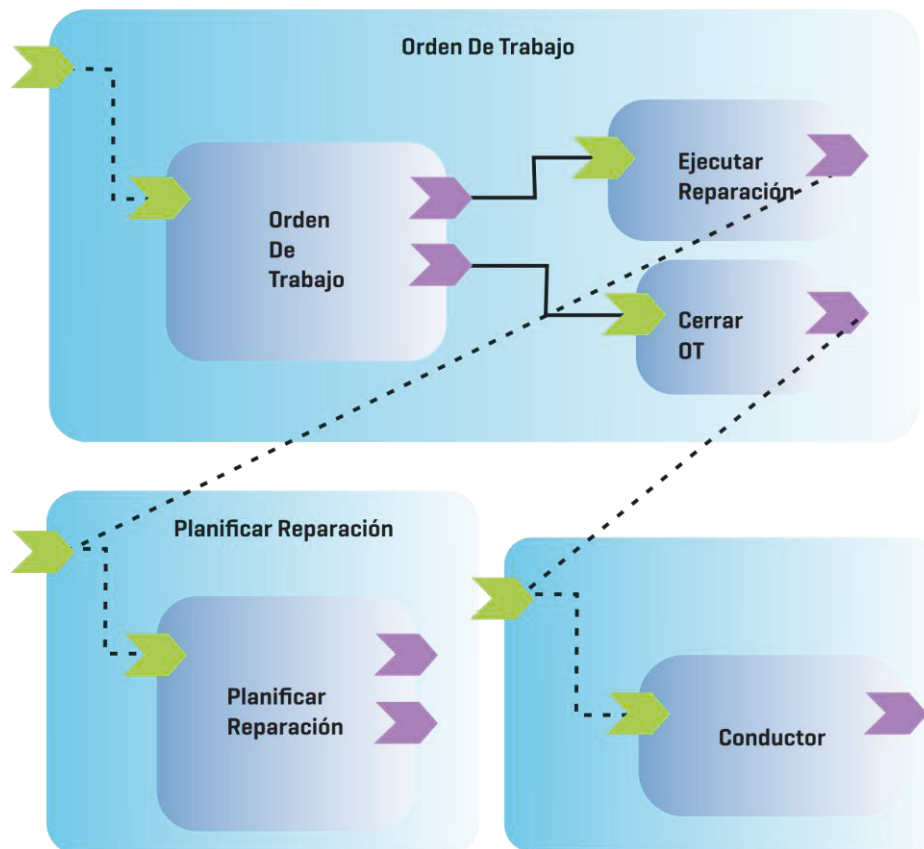
relaciones con aquellos que quieren usarlos. Concluida la etapa de identificación y modelado de servicios, continúa la etapa de definición de componentes.



**Figura 26. Prototipo de un Editor Gráfico de Servicios**

En esta etapa, se empaquetan los servicios como componentes de granularidad gruesa, para dar respuesta a los requisitos funcionales y no funcionales identificados en etapas iniciales y que son utilizados por procesos de negocios que los requieren.

En la Figura 30, se observan los componentes SCA identificados en nuestro caso de estudio. Si bien varios de los componentes se vinculan directamente a componentes SCA, cabe destacar que los enlaces SCA permiten representar también el flujo de trabajo subyacente. Esta figura fue construida con una herramienta para SCA para Eclipse [116].



**Figura 27. Modelo de Componentes SCA para el Ejemplo**

Este modelo de componentes surge de aplicar al modelo de servicios construido previamente, la generalización definida y las reglas de integración presentadas en la Tabla 3. En este caso, cada *ConceptualService* de P2S (*Service* en el metamodelo original P2S), se representa como un *Component* en SCA. Los Contratos son *Wire* y los ElementosTipados, *Reference*.

#### 4.5.5. Discusión

Se ha planteado un ejemplo para ilustrar la aplicación de MISP mejorada por la interacción entre el modelado de procesos, el modelado de servicios y la definición de componentes utilizando las reglas definidas para tal interacción.

Se observa que la definición de un lenguaje para representar el modelo de servicios permite formalizar la tarea de análisis bottom-up que se lleva a cabo luego del método top-down que se aplica al modelar los procesos de negocio. Este análisis se refleja en un modelo de servicios, que documenta la vinculación entre ellos y las actividades de los procesos y clasifica dichos servicios para luego poder mapearlos directamente a componentes funcionales representadas en el estándar SCA.

La definición de P2S, la implementación de un prototipo de editor gráfico y la definición de las reglas de interacción entre BPMN, P2S y SCA, contribuyen a mejorar MISP aportando métodos y técnicas para achicar la brecha entre el negocio y la tecnología, dejando a su vez documentación de la fase de modelado de servicios y que constituye el producto de dicha fase.

## 4.6. CONCLUSIONES DEL CAPÍTULO

En este capítulo se presentaron las mejoras introducidas a MISP desde el punto de vista metodológico con el fin de enriquecer e integrar las etapas de modelado de procesos y servicios mediante reglas de transformación entre metamodelos. Para ello se define un lenguaje de modelado de servicio propio denominado P2S como paso intermedio entre el metamodelo BPMN y el metamodelo de SCA. SCA posee una adecuada especificación de componentes de fácil integración con el metamodelo de WSDL.

Si bien es cierto que WSDL es simplemente un lenguaje de descripción de Servicios Web y que éstos últimos son solamente una manera de implementar servicios en una arquitectura SOA, nuestra propuesta se sustenta en el hecho de que al proponer un enfoque metodológico integrador que cuenta con una etapa de modelado de servicios con P2S, es en esta instancia en que pueden descubrirse la existencia de piezas de software existentes para ser consumidas por las actividades de un proceso. Este activo de software puede ser encapsulado fácilmente como Servicio Web, aunque no haya sido concebido como componente SCA.

La formalización propuesta de los pasos de transformación e interacción entre cada una de las etapas de MISP contribuyen a un enfoque metodológico que logra una combinación balanceada del uso alternado de técnicas de desarrollo top-down para el desarrollo de proceso de negocios y bottom-up para el desarrollo de servicios.

Así, la metodología mejorada absorbe buena parte de los activos de software existentes en las organizaciones de hoy, incluso en aquellos casos en que el paradigma de solución no sea estrictamente orientado a servicios. Además permite incorporar el análisis bottom-up de las clases subyacentes a las actividades del proceso y representar las estructuras de datos que la notación BPMN no refleja.

Este capítulo presenta las mejoras a MISP en cuanto al modelado de procesos y servicios. En el próximo capítulo presentamos mejoras a las etapas de despliegue y ejecución de procesos con vistas a enriquecer el monitoreo de los mismos.

## CAPÍTULO 5

# BPMS SOCIALES, COLABORATIVOS Y DISTRIBUIDOS

Con el crecimiento vertiginoso de Internet en los últimos años, conceptos como Web 2.0, middleware, virtualización, computación grid (tecnología que permite utilizar de forma coordinada todo tipo de recursos, entre ellos cómputo, almacenamiento y aplicaciones específicas) que no están sujetos a un control centralizado y otras tecnologías han dado origen al concepto de computación en la nube, como un nuevo modelo de negocio que impacta directamente en la manera en que se conciben las aplicaciones. En este contexto, aparecen nuevos desafíos técnicos que conllevan nuevos enfoques de utilización de aplicaciones por parte de sus destinatarios finales, los usuarios, que incorporan nuevos comportamientos de uso a los cuales el software debe dar respuesta.

En este capítulo se analiza el impacto de estos conceptos en la metodología MISP y se propone una solución que incorpora alternativas para afrontar estos nuevos desafíos tanto técnicos como de uso de las aplicaciones. Primero, se presenta una motivación y visión general para comprender la necesidad de introducir criterios de distribución de procesos, que acompañen los nuevos modelos computacionales en la nube, como así también mejorar los rastros de ejecución de los procesos enriqueciéndolos con los nuevos modos de colaboración entre usuarios que introducen aspectos sociales en las aplicaciones. Estos representan avances tecnológicos significativos para ser incorporados a la gestión de procesos de negocios (Sección 5.1). Segundo, se introduce la solución propuesta en torno a incorporar el concepto de monitoreo distribuido en los BPMS (Sección 5.2). Tercero, se explica el prototipo desarrollado que implementa la solución propuesta incluyendo características de software social para la etapa de ejecución de procesos de negocio y el uso de sus rastros para enriquecer el monitoreo de los procesos (Sección 5.3). Finalmente, se discuten conclusiones (Sección 5.4).

### 5.1. MOTIVACIÓN Y VISIÓN GENERAL

En esta sección se discuten los procesos de negocio como servicio - Business Process as a Service, BPaaS (Sección 5.1.1), la administración de procesos de negocio en la nube (Sección 5.1.2) y las características más relevantes de computación en la nube en el contexto de BPM (Sección 5.1.3).

#### 5.1.1. BPaaS - Business Process as a Service

La integración entre procesos de negocios y computación en la nube continúa siendo vaga en la actualidad. Si bien la industria predice una gran inversión de recursos en ese sentido y es evidente la necesidad de integrar ambas áreas, no está claro aún de qué manera hacerlo. Por otra parte, los problemas actuales con

los procesos de negocio automatizados radican en la pobre conexión entre las aplicaciones y los procesos, así como la existencia de mucha lógica embebida en el código de las aplicaciones.

A pesar de las inversiones de recursos que se vienen realizando en torno a la innovación de la tecnología de procesos, muchas aplicaciones continúan ejecutándose dentro de centros de datos corporativos y encapsulan los procesos de negocios dentro de sus arquitecturas de software. Llevar estas aplicaciones a la nube no resuelve la situación sino que la magnifica dado que dificulta aún más la conectividad con los BPMS que continúan bajo la modalidad *on-premise* [125].

Es importante distinguir el concepto de arquitectura de computación en la nube de arquitectura de aplicaciones con computación en la nube. El primer concepto se refiere a la organización conceptual a alto nivel de la plataforma que hace posible la computación en la nube sin considerar las tecnologías subyacentes. El segundo se refiere a la estructura de las aplicaciones que se ejecutan en un entorno donde el proceso es prestado como un servicio (PaaS - Process as a Service) y eventualmente a las aplicaciones basadas en el software como servicio (SaaS) que integran servicios de diferentes proveedores. La propuesta de solución tecnológica presentada en este capítulo se enfoca en el segundo concepto.

Dentro de un enfoque de orientación a procesos, la externalización de procesos, en Inglés llamada BPO (BPO - *Business Process Outsourcing*), dentro de un enfoque de orientación a procesos constituye sin dudas un cuarto modelo de servicio en la nube (junto con PaaS, SaaS y IaaS) la nube denominado BPaaS y que requiere de métodos y técnicas que lo definan.

En la actualidad, lamentablemente, los términos BPaaS y SaaS (SaaS - Software as a Service) aplicados a la ejecución automática de procesos de negocios, suelen confundirse entre sí. La mejor definición de BPaaS es provista por IBM, ya que fusiona la visión de dos importantes referentes de la industria – Gartner e IDC [126]:

#### **Definición 10. BPAAS – PROCESO COMO SERVICIO**

*Proceso como servicio (BPaaS) es cualquier proceso de negocio (horizontal o vertical) desplegados en el modelo de servicio de Cloud (multi-inquilino, autoaprovisionamiento de servicios, escalabilidad elástica y pago por uso) vía Internet, con acceso vía interface Web y explotando la arquitectura cloud orientada a Web*

La definición menciona procesos de negocios verticales y horizontales. Los procesos verticales son los que se limitan a integrar aplicaciones independientes (como por ejemplo, una aplicación de sueldos y jornales, la administración de clientes o *Cestones Rezurce Magnamente* (CRM), y una aplicación de Recursos Humanos) que son orquestadas por un proceso. Los procesos horizontales son aquellos procesos y que utilizan servicios o aplicaciones que se ejecutan dentro de un BPMS y que llevan la funcionalidad del BPMS a la nube.

Conceptos un poco más avanzados se presentan cuando los requisitos para considerar un entorno Bas incluyen alguna variante de sindicación de procesos y su disposición en un intermediario (broker). La sindicación - un medio para notificar y actualizar contenidos compartidos - permitiría la creación dinámica de procesos de negocios, de manera análoga a como se crean instancias de máquinas virtuales, e incluiría el concepto de UDDI (UDDI - *Universal Description Discovery and Integration*) que es parte del protocolo de los servicios Web [127].

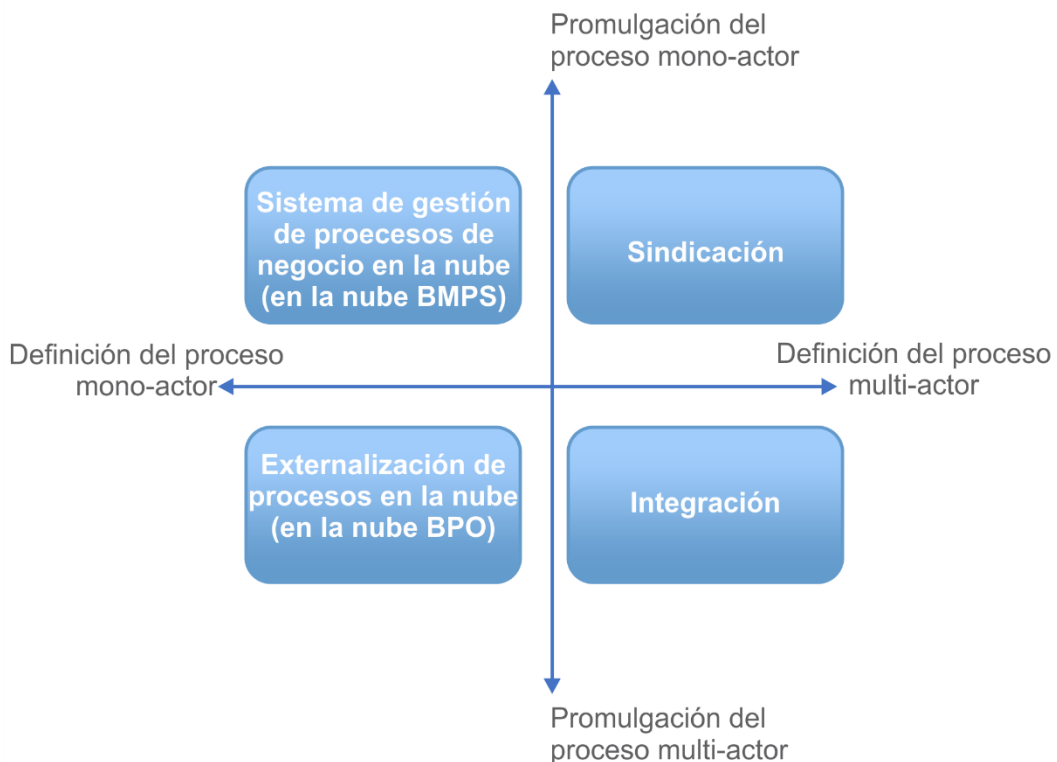
Basado en estas características, en [125] se define:

#### **Definición 11. PLATAFORMA EN LA NUBE DE PROCESOS DE NEGOCIO**

*Plataforma en la nube de Procesos de Negocios es una plataforma compuesta por servicios en la nube que habilitan la definición, promulgación y sindicación de procesos de negocios (completamente automatizados o soportados por las personas) tanto horizontal como verticalmente vía mecanismos basados en Web o APIs.*

Esta definición se basa en la interpretación de IBM para BPaas [126] pero agrega algunas propiedades que marcan la dirección de los procesos de negocios con una interacción humana cada vez más parcial y reducida.

La Figura 28 muestra distintos escenarios para la ejecución de procesos de negocios y la forma más conveniente para su ejecución y definición dependiendo si el proceso es ejecutado por uno (mono) o varios (multi) actores



**Figura 28. Escenarios Posibles**

En la Externalización de Procesos de Negocios en la Nube (Cloud BPO), el propietario de la definición del proceso es el proveedor del mismo o una tercera parte que provee la definición de procesos como un producto. En este caso el cliente del proceso de negocios solo realiza la configuración del mismo (promulgación). En el caso de Sistemas de Gestión de Procesos de Negocios en la Nube (Cloud BPMS), el cliente del proceso es el propietario de la definición y de la promulgación del proceso; exactamente como en el caso de BPMS tradicional. En la Sindicación de procesos de negocio, el cliente promulga dos o más procesos de un único propietario. Existe una definición de procesos intermedia definida por el cliente pero el sub-proceso real tiene otro propietario.

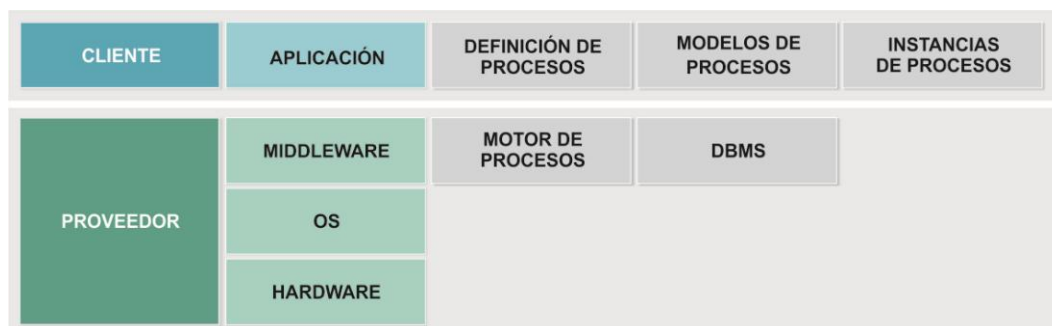
Finalmente, en la Integración de procesos de negocios se permite realizar procesos colaborativos en entornos de negocios (Business to Business - B2B). Esta última alternativa está fuera del alcance de esta propuesta

### 5.1.2. BPM en la Nube

Habiendo analizado el enfoque de procesos de negocio como servicio (BPaaS), a continuación se describen tres posibles escenarios para la Gestión de Procesos de Negocio en la Nube, según los modelos de servicios clásicos. Los escenarios también muestran la manera en que se distribuyen las componentes de un BPMS. En particular, la componente “cliente” representa a la organización propietaria del BPMS y de los procesos que allí se despliegan (on-premise) (ver Figuras 29 a 31); mientras que la componente “proveedor” representa el proveedor del servicio en la nube, pudiendo ser este público o privado.

Las tres variantes de Gestión de Procesos de Negocio en la Nube incluyen:

**Infraestructura como Servicio** - Infrastructure as a Service (IaaS) - en este modelo de servicio, ningún componente del BPMS se encuentra en el proveedor de la nube, quien solamente da soporte de hardware. Como muestra la Figura 29, el proveedor de la nube presta los servicios de hardware, sistema operativo (SO) y de middleware, que puede incluir el motor de procesos y el sistema de administración de base de datos, entre otros. El cliente que necesita ejecutar una aplicación, es propietario de la definición de procesos, como así también de los modelos y las instancias.

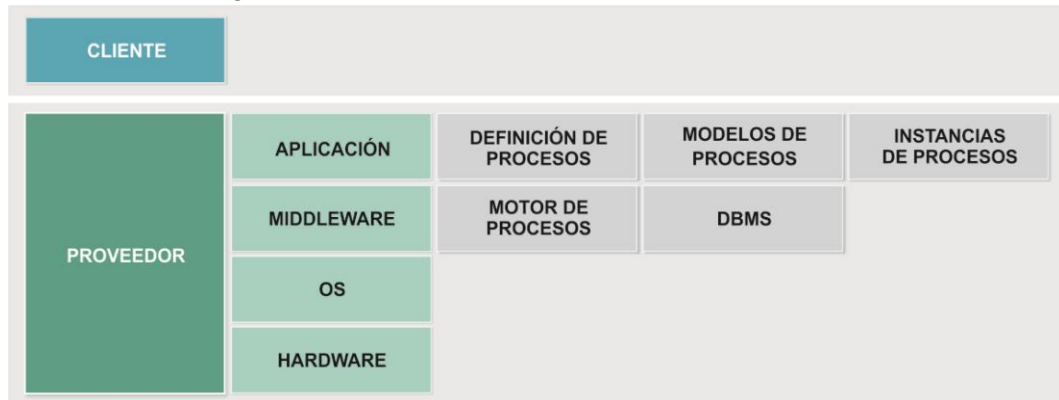


**Figura 29. BPMS como IaaS**

**Plataforma como Servicio** - Platform as a Service (PaaS) - en este modelo de servicio, se despliegan en el proveedor de la nube únicamente el motor de

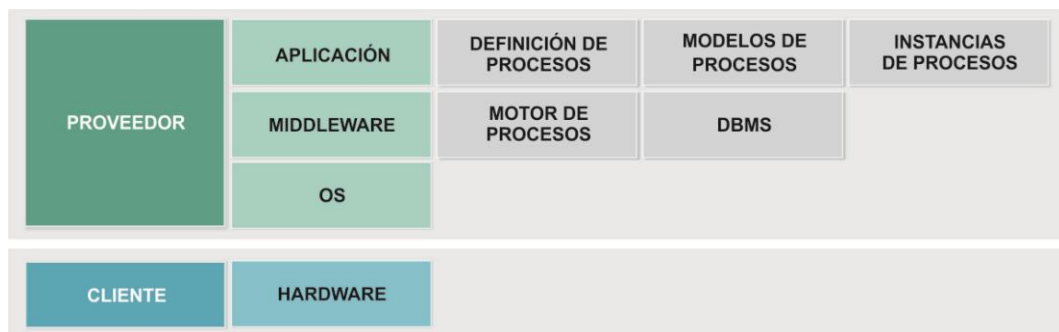


procesos y el motor de base de datos, quedando dentro del ámbito del proveedor, en modalidad on-premise, las definiciones de procesos, sus modelos y sus instancias (ver Figura 30).



**Figura 30. BPMS como PaaS**

**Software como Servicio** - Software as a Service (SaaS) - en este modelo de servicio, todas las componentes del BPMS se despliegan en el proveedor de la nube (ver Figura 31). Este escenario la posibilidad de ejecución de la aplicación depende completamente de la disponibilidad de los recursos en la nube.



**Figura 31. BPMS como SaaS**

### 5.1.3. BPM en la Nube: características del modelo

A continuación se explican tres características relevantes del modelo de Gestión de Procesos de Negocio (BPM) en la nube.

#### Arquitectura Orientada a Servicios – Service Oriented Architecture (SOA)

La computación en la nube provee servicios que aportan transparencia al usuario – el usuario se abstrae de problemas específicos, como por ejemplo, capacidad de almacenamiento. La solución tecnológica propuesta en esta tesis, basada en la metodología MISIP [1] ha considerado ampliamente la vinculación de BPM con SOA. En particular, los servicios como componentes a consumir por las actividades de un proceso, conforman la base fundamental de dicha metodología. En el Capítulo 4 se presentó la propuesta de mejora a MISIP introduciendo la noción de conceptualización de servicios como paso previo a su implementación. Por lo tanto, la metodología está basada en principios de arquitectura orientada a servicios

## **Basado en componentes, extensible y escalable**

El sistema de instancias de la computación en la nube está conformado con la integración funcional de componentes. La escala de ejecución de los recursos puede crecer de acuerdo a las necesidades de manera automática. Este concepto aplicado a BPM se traduce en el concepto de procesos distribuidos que puedan alojarse a demanda, pero sin perder la visión del proceso completo. Esta es la propuesta que se plantea en este capítulo.

## **Multi-inquilino**

El concepto de multi-inquilino no sólo se limita al modelo IaaS, en el modelo PaaS, se espera contar con servidores propios para cada inquilino como Application Servers, JVM e incluso Process Server. En nuestra propuesta, esta es una pieza fundamental, habida cuenta que se requieren diferentes motores de procesos para ejecutar cada parte de un proceso distribuido.

## **5.2. PROCESOS DISTRIBUIDOS EN BPMS: SOLUCIÓN PROPUESTA**

En este capítulo se presenta la solución propuesta a la distribución de procesos y su despliegue en BPMS distribuidos. En la Sección 5.2.1 se introduce la motivación, luego se plantean las problemáticas de la ejecución distribuida de procesos (Sección 5.2.2) y en la Sección 5.2.3 se muestra el monitoreo de procesos en la nube y su prototipo.

### **5.2.1. Motivación**

Dentro de las mejoras propuestas a MISIP en esta tesis, una de ellas se refiere al análisis del impacto que la computación en la nube puede tener en dicha metodología y una propuesta de solución.

La adopción de un BPMS en una organización puede constituir un proyecto costoso tanto desde el punto de vista de la inversión que debe realizarse en software y en hardware, así como en los recursos humanos entrenados para desplegar y mantener la solución. Por otra parte, la escalabilidad puede ser un aspecto a considerar en estas organizaciones dado que el motor de procesos que se adopte será capaz de atender simultáneamente una cantidad limitada de instancias de procesos [145].

En este sentido, la necesidad de escalar y la naturaleza netamente colaborativa que subyace a la ejecución de procesos de negocio, ponen en juego la necesidad de contar con entornos descentralizados que permitan optimizar el uso de la tecnología y dar respuesta a la necesidad de crecimiento de las organizaciones. Es así que las arquitecturas basadas en la nube ofrecen una buena alternativa por su enfoque flexible y ágil. En particular, los procesos de negocio colaborativos en los cuales interactúan distintos sectores de una o de diferentes organizaciones, ofrecen un potencial de optimización a través de la combinación de técnicas de computación en la nube y de BPM.

La solución propuesta en esta tesis aborda solamente una de las etapas del ciclo de vida de los procesos que es la de evaluación o monitoreo y que se representa en MISIP como la “Etapa 8-Administración y Seguimiento”. Esta etapa favorece el seguimiento de los procesos de negocio desde la Etapa 1 y que contribuye a cerrar el ciclo de mejora continua en un entorno distribuido.

Si bien la distribución de procesos presenta como prerrequisito la descomposición de los procesos, este tema ha sido ampliamente abordado en los últimos años. Sin embargo, el monitoreo de los procesos de negocio que se ejecutan bajo un esquema de descomposición, no ha sido demasiado explorado aún.

Este trabajo propone una implementación para una arquitectura de sistema de monitoreo de procesos distribuidos utilizando Bonita Open Solution [133] – una suite para la gestión de procesos de negocios open source - como motor de procesos, su API y el uso de conectores personalizados.

### **5.2.2. La Ejecución Distribuida de Procesos de Negocio**

Para beneficiarse de una arquitectura distribuida de procesos surge la necesidad de dividir los procesos de negocios en procesos colaborativos individuales que puedan ejecutarse indistintamente de manera embebida o en la nube. Esto conlleva ciertos problemas en la ejecución de procesos, como por ejemplo la descomposición, la sincronización y el monitoreo [130] [144] [145], los cuales se describen a continuación:

#### **Descomposición de Procesos**

Al disponer procesos en la nube surge la inquietud de qué hacer con la gestión de información sensible y como evitar publicar bases de datos corporativas en un entorno externo. La descomposición de procesos aparece necesaria en estos esquemas híbridos donde algunos procesos pueden externalizarse y otros no. Así parecería natural que las actividades que requieran más acceso a la información permanezcan dentro de la organización evitando su publicación y el tráfico que se generaría para accederla.

#### **Sincronización de Procesos Distribuidos**

Un proceso distribuido (o particionado) se divide de acuerdo a la cantidad de servidores disponibles. Los servidores donde se ejecutan deberán contar con algún mecanismo de sincronización cuyas soluciones no son muy diferentes a las distintas variantes de sincronización en cualquier entorno distribuido, siendo este un campo altamente explotado en el ámbito de las bases de datos. A fin de implementar dicha sincronización, es necesario contar con un mecanismo de notificación entre las actividades de los procesos. Las notificaciones se implementan usando colas de mensajes y un proceso en segundo plano (tipo demonio) para reunirlos (*pooling*). Este demonio recibe mensajes e inicia instancias de los procesos requeridos. De este modo, cada servidor en el modelo híbrido (los embebidos y también los basados en la nube) deben contar con una copia del servicio de *pooling* para recibir las notificaciones de finalización y luego notificar al motor de procesos. El resultado de esto es la capacidad de iniciar cada instancia de los procesos según la definición de cada uno de ellos [128] [129].

## Monitoreo de Procesos Distribuidos

Dos problemas principales al distribuir o particionar procesos son el monitoreo de las distintas instancias y el mantenimiento de la visión integrada del proceso completo. Si bien las mejoras del proceso se aplicarán al proceso real, como un todo, la distribución es únicamente para mejorar el rendimiento del proceso y escalar en cuanto al número de usuarios y facilitar la accesibilidad de manera remota. Para lograr este objetivo será necesario asociar a cada instancia de proceso con su proceso original y luego contar con un aplicativo que recupere los rastros y permita visualizar el proceso completo, siendo transparente para el usuario la distribución del proceso.

Esta funcionalidad se puede obtener mediante el desarrollo de una aplicación web en la nube capaz de acceder a cada uno de los servidores distribuidos – ya sea en la nube o embebidos. Para ello la aplicación deberá tener un catálogo de los servidores existentes en la arquitectura y en cada uno de ellos un servicio web que reciba el identificador de una definición de proceso y retorne la información de cada instancia asociada a dicha definición, conteniendo: el identificador de la instancia del proceso, el estado – por ejemplo, en ejecución, completo, o suspendido; la actividad actual en caso de no estar completo, la fecha y hora de ejecución. Con esto, la aplicación web de la nube envía una invocación al servicio web por cada definición de proceso y cuenta con la información para mostrar cada instancia particular de los procesos y sus detalles. La información retornada incluye el identificador de la actividad, el participante asociado, fecha de comienzo, estado actual y fecha de finalización.

Después de recibir esta información, la aplicación web permitirá al usuario observar los detalles de las actividades de manera transparente sin indicar cuál fue el servidor que la ejecutó otorgando la visión completa del proceso definido, instanciado y ejecutado [129] [131] [132].

### 5.2.3. Monitoreo de Procesos en la Nube

Como hemos mencionado, dos principales problemas al particionar el modelo de procesos es reunir y monitorear diferentes instancias distribuidas (se dispongan éstas de manera embebida o en la nube) y lograr una visión integrada bajo la óptica del proceso original al que pertenecen las instancias. Para ello se diseña una solución que considera componentes distribuidas e interconectadas formando una arquitectura que describimos a continuación.

A fin de soportar la distribución, será necesario asociar las diferentes instancias de procesos iniciadas dentro de una cadena con el objetivo de reunir la información acerca del acceso a los diferentes servidores de procesos. El modelo de ejecución de procesos descompuestos consiste en enlazar cada flujo de instancias con el proceso particionado correspondiente.

Así, cuando una instancia finaliza en un servidor, inicia automáticamente una nueva instancia correspondiente a la siguiente partición del proceso, dependiendo de la arquitectura de distribución.

Para ello, cada nodo de la arquitectura debe establecer comunicación con el nodo siguiente para iniciar la nueva instancia y reunir la información de ambas. Es decir,

cuando una nueva instancia se inicia en un nodo de la arquitectura, debe obtener no solamente sus datos sino cada instancia generada en el otro servidor.

A continuación presentamos una solución de software que representa un Sistema de Gestión de Procesos de Negocio (BPMS) sobre la cual implementaremos la solución propuesta para la distribución de procesos.

### Bonita Open Solutions: API y conectores

Existen varias alternativas para implementar flujos de instancias de procesos enlazadas. En nuestro caso elegimos Bonita Open Solution [133] como el BPMS. De este modo, una vez que el proceso original es particionado y cada parte es desplegada en el servidor correspondiente, siguiendo un criterio que favorezca la sensibilidad de los datos, el tráfico de transferencia y la portabilidad de las aplicaciones involucradas en las diferentes actividades del proceso, se utiliza la API (API – *Application Programming Interface*) de Bonita y sus conectores para crear instancias y recuperar la información utilizando clases Java.

Las clases a programar en Java usan la API de Bonita como librerías incluyendo el acceso a funciones que permiten la autenticación del servidor, definir la instancia lanzada, obtener información de la instancia e inicializar las variables del proceso. Estas clases son invocadas desde la definición del proceso usando conectores.

Un conector en Bonita está dividido en dos partes, la definición y la implementación. La definición del conector controla las interfaces externas del conector, es decir, aquellas visibles al usuario y aquellas visibles al motor de Bonita. La implementación del conector consiste en un archivo XML que describe identificadores, versiones y dependencias del conector, y una clase Java que extiende la clase *ProcessConnector* (una clase abstracta), en la cual el creador del conector escribe el código para el nuevo conector en un IDE (IDE - *Integrated Development Environment*) básico similar a Eclipse [135].

Bonita provee un asistente para crear conectores paso a paso de muy simple uso, como se muestra en la Figura 32.

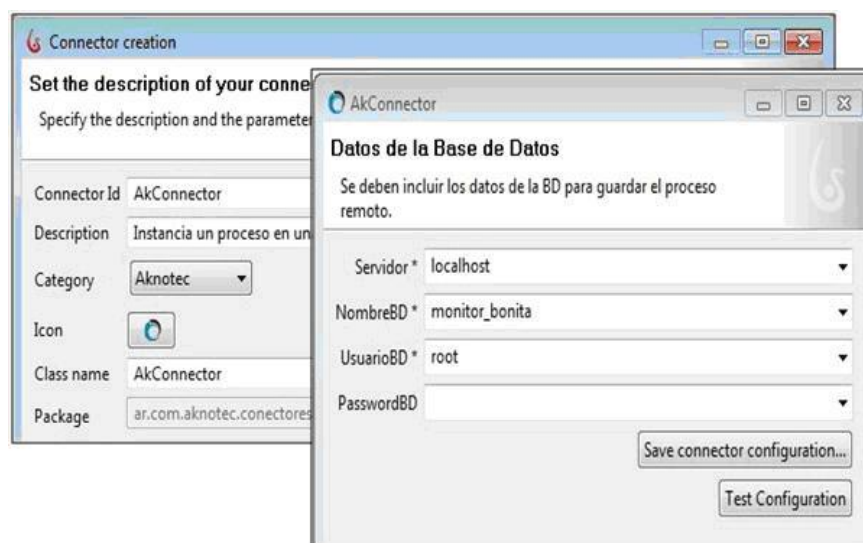


Figura 32. Asistente para la Creación del Conector en Bonita

En cada proceso se incluye la información necesaria para la comunicación con otros servidores Bonita dentro de la arquitectura, y en este caso, también mediante conectores, se puede lanzar nuevas instancias en cada servidor. Así, cada vez que finaliza una instancia, se activa el conector que inicia una nueva instancia lo cual permite continuar con el flujo del procesos distribuido y obtener la información necesaria de la instancia remota [[134] [84].

A fin de implementar la solución propuesta para la ejecución de procesos distribuida, definimos un conector en Bonita. La tarea de nuestro conector consiste en comunicarse con otro motor de ejecución de Bonita situado en un servidor remoto, crear una instancia de proceso en ese servidor, enviar aquellos datos necesarios como entrada a la nueva instancia, y guardar los identificadores de las instancias locales y remotas en una base de datos para luego poder relacionar ambas instancias al momento de realizar el monitoreo.

Para instanciar un proceso situado en otro servidor, es necesario que el conector, localizado en el servidor local, se ejecute en el servidor remoto. Esto es posible realizando un cambio de contexto para que el conector conozca la dirección IP del servidor remoto, así como también sus parámetros de acceso (usuario y contraseña). Este cambio de contexto y la autenticación al servidor se realizan a través de configuraciones JAAS [136]. Previo a esto se debe suministrar al conector el nombre del proceso remoto y su versión, para poder así instanciar el proceso y continuar con la ejecución. Una vez creada la instancia en el servidor remoto, los datos correspondientes tanto de la instancia local como remota, se almacenan en una base de datos local que se utilizará para realizar el seguimiento de todo el proceso. En nuestra solución se plantea el uso de una base de datos MySQL, por lo que se requerirá importar el manejador de Java (`com.mysql.jdbc.Driver`) para interactuar con la misma. Finalmente, el conector desarrollado debe restablecer el contexto en que se encontraba el sistema previo a su ejecución.

### Front-end Centralizado

Tal como se ha mencionado, la aplicación de monitoreo debe desarrollarse con el objetivo de mostrar los datos integrados de cada instancia distribuida. Frente a la ejecución enlazada, es importante que cada instancia pueda almacenar, no solo su propia información, sino la asociada a las instancias que ha creado en otros servidores. Así, al acceder a la instancia inicial del proceso, es posible recuperar la información de la próxima y así sucesivamente, para completar el proceso.

Recuperada esta cadena de ejecución en los diferentes servidores, se debe construir una aplicación para visualizar los datos recopilados y mostrarlos correctamente.

Lo fundamental es otorgar al usuario total transparencia de la distribución, por lo tanto la aplicación deberá acceder a cada servidor distribuido (ya sea embebido o en la nube). Para esto, se deberá contar con un catálogo de servidores existentes. Cada uno de ellos contiene una copia del servicio web (`getInstanceService`) que recibe una definición de proceso y retorna la información de cada instancia existente en el servidor. La información provista por el servicio web incluye: la

identificación del proceso, el participante, la fecha de inicio, el estado actual y la fecha de finalización.

Una vez recolectada toda la información se concatena la misma con la existente acerca de cada instancia de la cadena, obteniendo así el flujo completo. La información es visualizada en una aplicación web donde el usuario puede seleccionar una instancia en particular y observar sus detalles. Para ello, la aplicación utiliza otro servicio Web (*getInstanceActivityService*) para obtener de cada servidor los detalles de cada actividad.

Si bien la información del servidor donde se ejecutó la instancia no está visible, la aplicación cuenta con la información correspondiente pudiendo mostrarla a demanda del usuario facilitando el monitoreo del proceso [137] [138] [139].

### **Arquitectura de la aplicación**

La arquitectura de aplicación de monitoreo de procesos distribuidos que se propone [144] [145], muestra en la Figura 33 las diferentes componentes distribuidas así como las relaciones internas entre ellas y el usuario.

La solución se compone de tres nodos principales: 1) la nube – *Cloud Component*, 2) el embebido o sistema tradicional – *Embedded Component*, y 3) la aplicación de monitoreo – *Monitoring Component*. El *Cloud Component* trabaja como contenedor de varios elementos: el BPMS, la aplicación de monitoreo, la API REST (REST - *Representational State Transfer*) utilizada para integrar la aplicación con los motores de procesos y eventualmente un servicio de geolocalización que permite asignar clientes móviles para versiones más convenientes del servicio.

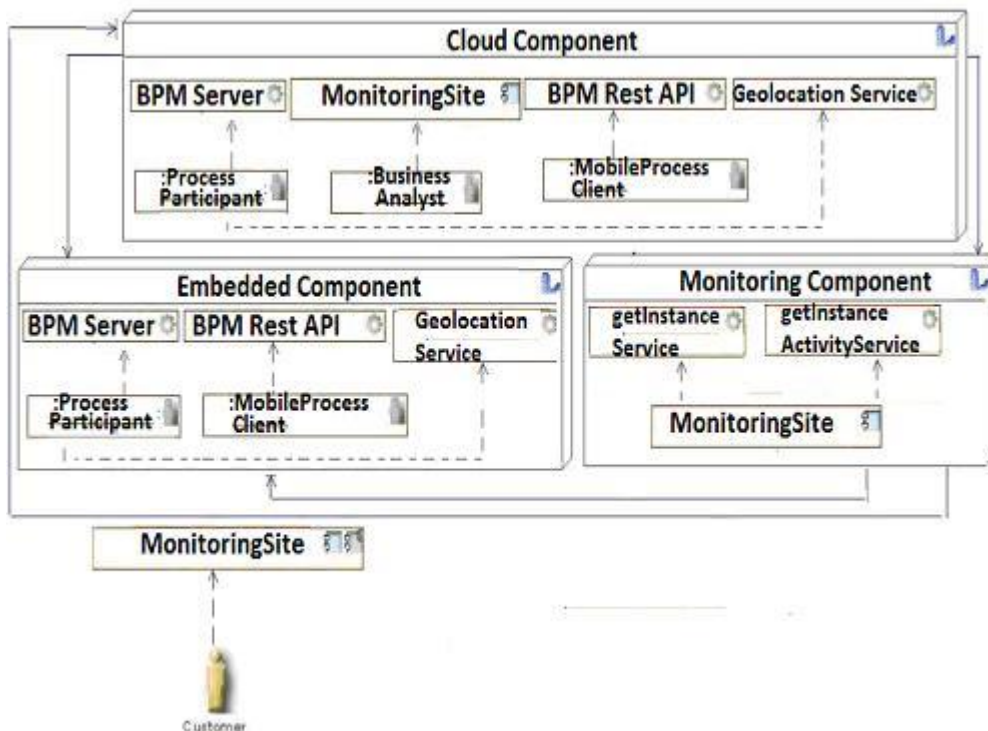
El *Embedded Component* incluye componentes embebidas de aplicaciones BPM tradicionales que pertenecen a la organización y que por distintas razones – sensibilidad de los datos o portabilidad de las aplicaciones – se decide no alojarlas en la nube. El comportamiento de *Embedded Component*, desde el punto de vista funcional, es equivalente al de *Cloud Component*.

El *Monitoring Component* provee la funcionalidad necesaria para el monitoreo de las instancias de procesos. Por ejemplo, devuelve información acerca de instancias y las actividades utilizando los servicios web *getInstance* y *getInstanceActivity*, que se comunican con servidores de procesos a través de la API de Bonita [77] [85] [88].

### **Comunicación de componentes**

La Figura 34 muestra el análisis de la comunicación entre los componentes de la arquitectura. Es importante observar las aplicaciones involucradas, los actores principales y la interacción de los diferentes componentes de software. Si bien el actor preponderante en los resultados del monitoreo es el participante de cada actividad, los analistas de negocios así como los administradores de la arquitectura, son actores involucrados que pueden optimizar los servicios o los componentes del proceso. La figura también muestra como los componentes de ejecución y de monitoreo pueden accederse indistintamente en la nube o en nodos embebidos [83] [81].

A continuación presentamos algunos detalles de la implementación de la solución propuesta.



**Figura 33. Arquitectura de la Aplicación de Monitoreo e Interacción del Usuario**

### Implementación de la Aplicación

La aplicación Web se desarrolló utilizando las tecnologías PHP, HTML y CSS y de las siguientes librerías complementarias [140]:

- JQuery y JTable: para una comunicación y diseño más agradable a la vista del usuario.
- NuSOAP: para la comunicación entre las aplicaciones de monitoreo y el motor de Bonita [141]
- SimpleXML: como estándar para el formato de la información de las instancias [142]
- GraphViz: para diagramar el proceso de negocio distribuido unificado.

La solución desarrollada permite monitorear cada una de las instancias que han sido ejecutadas desde el conector, mostrando además de la instancia local y la instancia remota, cada una de las actividades involucradas en las mismas, obteniendo información detallada de las actividades del proceso, incluyendo: el estado, fecha y hora de inicio y, finalización, el actor, etc. La Figura 35 muestra un ejemplo de la visualización de la información de monitoreo.



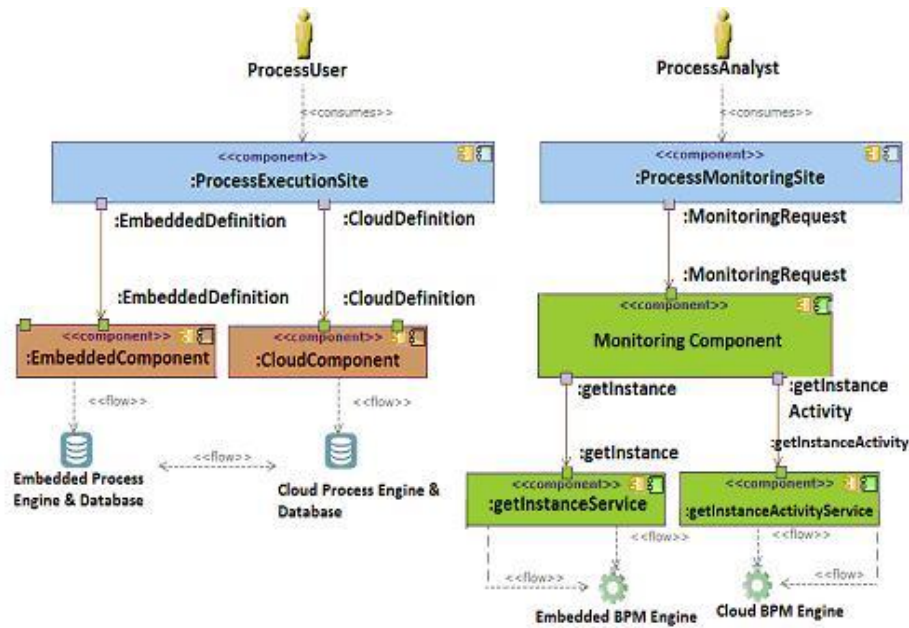


Figura 34. Diagrama de Comunicación de la Aplicación

Tabla de instancias						
Instancia Local				Instancia Remota		
Ping--1.2--1				GatewayPong--1.1--42		
Información del proceso: Ping--1.2--1						
Label	Estado	Ready	Inicio	Fin	Tipo	
Start1	FINISHED	8/10/2013, 10:15:52	8/10/2013, 10:15:52	8/10/2013, 10:15:52	Automatizado	
Instanciar Gateway Ping en Server 1	FINISHED	8/10/2013, 10:15:52	8/10/2013, 10:16:08	8/10/2013, 10:16:08	Humano	
Ver datos de la solicitud	FINISHED	8/10/2013, 10:16:35	8/10/2013, 10:16:50	8/10/2013, 10:16:50	Humano	
End1	FINISHED	8/10/2013, 10:16:50	8/10/2013, 10:16:50	8/10/2013, 10:16:50	Automatizado	
Ping--1.2--10				GatewayPong--1.1--50		
Ping--1.2--11				GatewayPong--1.1--51		
Ping--1.2--12				GatewayPong--1.1--52		
Ping--1.2--13				GatewayPong--1.1--53		

Figura 35. Visualización de las Instancias y Actividades

Finalmente, para obtener una visión unificada del monitoreo distribuido se genera un archivo XML que consolide cada uno de los conjuntos de información obtenidos de las instancias remotas que permiten identificar los tipos de actividades que se encuentran en las instancias, es decir, si son tokens de inicio o de fin, tareas, compuertas, flujos, si tienen conectores, actores, etc. Una vez conocida la información para generar el proceso original, se utiliza la librería GraphViz [144], que nos permite graficar las actividades como nodos y el flujo de control como aristas direccionales logrando una visualización del proceso de negocio original similar a la que puede realizarse con la aplicación de Bonita Studio (ver Figura 36).

Así, el usuario encargado de realizar el monitoreo de los procesos, consigue un panorama más perceptible del proceso completo, pudiendo determinar con mayor rapidez las tareas que deberían reconsiderarse para obtener una mayor eficiencia en la totalidad del proceso de negocio.

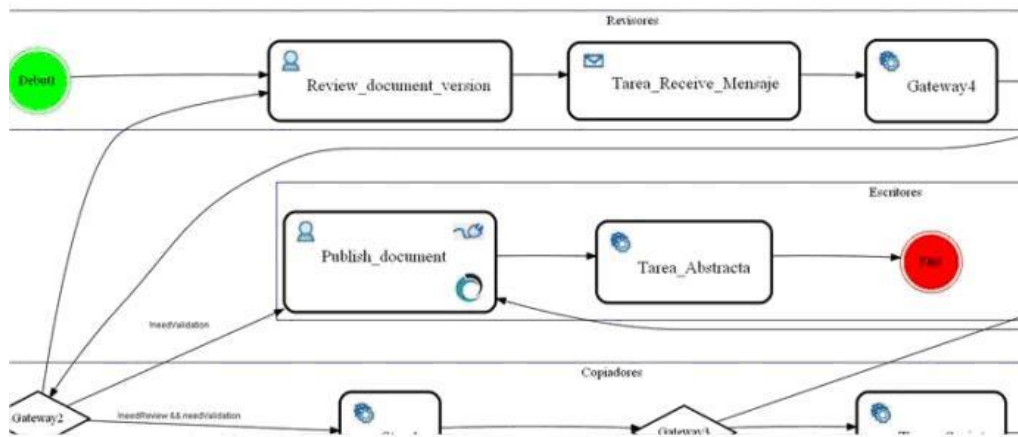


Figura 36. Visualización en GraphViz

### 5.3. BPMS SOCIALES Y COLABORATIVOS: SOLUCIÓN PROPUESTA

En esta sección se describe la solución propuesta para incorporar a los BPMS características sociales y colaborativas. Se presenta SocialFlow, un prototipo de *workflow* social, construido para validar la solución desarrollada.

#### 5.3.1. Motivación

Los procesos de negocios pueden requerir de entornos colaborativos y distribuidos. Ejemplos de este tipo de procesos son los aquellos ejecutados por varias agencias de gobierno para la prestación de servicios públicos. A diferencia de los procesos de negocio ejecutados por una sola organización, los procesos colaborativos y distribuidos requieren incorporar las siguientes funcionalidades [20]:

*Interacción Global* – poder recolectar, ofrecer e intercambiar información que pueda compartirse por varios actores, cada uno de ellos responsable por la ejecución de una o más tareas del proceso.

*Inteligencia Colectiva* – permitir intercambiar, sistematizar y evaluar información entre los actores

*Evaluación Permanente* – recibir retroalimentación del comportamiento del usuario para mejorar en forma continua y permanente

*Interoperabilidad* – facilitar la recolección y la provisión de información independientemente de la plataforma tecnológica de los actores y de manera ubicua.

A fin de entregar estas funcionalidades, las aplicaciones que ejecutan los procesos (BPMS) deben considerar nuevos conceptos así como también deben mejorar el enfoque para el modelado de los procesos. A continuación se resumen algunas recomendaciones para los BPMS sociales y colaborativos:

*Socializar Procesos* - en muchos BPMSs, los usuarios cuentan con una visión limitada de los procesos en los que participan, perdiendo información de contexto como por ejemplo, las personas involucradas en el proceso y las historias y resultados de ejecuciones anteriores.

*Atomizar las Actividades* - en muchos diseños de procesos, las actividades se definen con una granularidad muy baja (actividades de alto nivel), lo que implica un esfuerzo sustancial por parte de los participantes para completarla. Aumentar dicha granularidad y en consecuencia atomizando las actividades (actividades más refinadas), alienta la participación de los actores y permite redefinir circuitos y dividir tareas.

*Utilizar un Esquema de Recompensas* - muchos procesos organizacionales definen sus actividades de manera vertical, es decir respetando líneas jerárquicas y basados en autoridad, y los participantes las ejecutan solo para cumplir con sus superiores. En el software social, la participación es voluntaria y el compromiso se establece obteniendo puntos de recompensa que pueden luego traducirse en reconocimientos informales o reembolsos *monetarios pero que incluyen al participante de manera participativa e incluso lo alienta a mejorar el proceso.*

A fin de poder ejecutar procesos sociales y colaborativos en una organización se requiere de dos elementos claves – 1) identidad digital y 2) confianza y reputación. Los mismos se explican a continuación.

*Identidad Digital* - la identidad de quien gestiona la información en un sistema es aplicable tanto en el software social, como en un arquitectura orientada a servicios, como en una Wiki o un blog, La identificación de la identidad de quien crea datos permite que el proceso identifique el usuario humano con la actividad que se está ejecutando.

*Confianza y Reputación* – una vez asumida la existencia de la identidad digital, es importante conocer las implicancias de la confianza y la reputación de las personas. En particular, el software social requiere proveer información útil para el contexto, vinculando a los usuarios unos con otros para recabar información del entorno que alimenten a los procesos y permitan gestionar las excepciones a los procesos de negocio.

A fin de ilustrar estos conceptos, la próxima sección presenta una especificación de requisitos para SocialFlow, un prototipo de *workflow* social para la ejecución de este tipo de procesos.

### **5.3.2. Especificación de Requisitos para SocialFlow**

Una de las principales debilidades de los BPMS actuales es su incapacidad de absorber el dinamismo en la ejecución de los procesos, el cual está directamente vinculado a la interacción real que se produce entre los participantes y que forma parte del proceso mismo.

Se especifican a continuación las características funcionales que debe tener un motor de ejecución de procesos para incorporar la interacción entre participantes y enriquecer los rastros de ejecución de sus actividades.

### **Notificaciones**

Las notificaciones son avisos que se disparan ante determinados eventos dentro de la ejecución de un proceso. Las notificaciones pueden surgir por diferentes razones:

*La creación de un nuevo ítem de trabajo* - notifica al participante de la actividad sobre la creación de un nuevo elemento de trabajo a realizar

*Cambios de prioridad* - las actividades de los procesos pueden cambiar de prioridad de acuerdo al tiempo restante para la ocurrencia de su fecha límite. Cuando este cambio ocurre se notifica al usuario.

*Mensajes internos* - ocurren por intercambio de mensajes entre los usuarios de la plataforma

*Nuevas etiquetas* – notifica al usuario correspondiente cuando se agregan comentarios a las instancias de las actividades.

Estas notificaciones deberán ser disparadas a cada participante involucrado en el evento, indicando al mismo la prioridad que posee y dejando registro de su envío y de su recepción.

### **Comentarios etiquetados**

Cada actividad que se defina dentro de los procesos, debe poder clasificarse como una actividad comentada o no comentada. En el primer caso, se indicará que dicha actividad, al momento de su ejecución, podrá contener un comentario a modo de texto donde el participante podrá etiquetar o mencionar a otro participante involucrado en el proceso al que pertenece la actividad. Este comentario quedará registrado como parte de la ejecución de la actividad y será notificado al destinatario (Ver apartado Notificaciones). El registro de este comentario podrá ser eliminado solamente por quien lo realizó.

Respecto de las etiquetas admitidas en el comentario, las mismas pueden referir: el nombre de un participante, el nombre de una actividad perteneciente al proceso en ejecución, el nombre de un bifurcador o *gateway* perteneciente al proceso en ejecución o una palabra clave previamente definida como variable de proceso.

Este registro de etiquetas se incorporan a los rastros de ejecución del proceso y podrán ser luego consideradas en los reportes de monitoreo de los procesos ejecutados.

A su vez, en el caso de actividades comentadas, debe ser posible que ante su compleción, se notifique a todos los mencionados en los comentarios, que la misma fue completada, junto con su fecha de compleción.

### **Definición de tareas ad-hoc sin modelar**

Esta característica funcional, permite crear nuevas tareas no previstas en el modelado del proceso y cuyo alcance es la tarea que se está ejecutando y su tiempo de vida es hasta la compleción de dicha tarea. Se comportan como sub-tareas dinámicas, que dejan registro de actividades no conocidas en el modelado original y que surgen de la interacción del participante con su entorno.

Estas sub-tareas se incorporan a los rastros de ejecución de la actividad en el momento en que se completa la misma, no pudiendo visualizarse mientras la dicha actividad principal se encuentra en procesos de ejecución.

Al momento de obtener estadísticas de monitoreo de los procesos, el usuario puede elegir o no considerar dichas tareas ad-hoc, que contendrán la misma meta-información que las tareas incluidas en el modelado (fecha y hora de inicio, fecha y hora de finalización, participante involucrado). Las tareas ad-hoc no admiten comentarios etiquetados.

### **Notificaciones a los responsables de los procesos**

Esta funcionalidad permite establecer distintos tipos de retroalimentación comunicacional entre los participantes de los procesos y los responsables de los mismos. En este caso, la funcionalidad adquiere características diferentes a las notificaciones entre los participantes, debido a los distintos niveles de jerarquía que existe entre un participante del proceso y el responsable del mismo.

En este sentido, la notificación deberá incluir tanto el aviso de compleción de la actividad como también el historial de conversaciones entre participantes, la creación de tareas ad-hoc y, eventualmente, debe poder solicitarse la excepción para ejecutar la tarea, por parte del participante, junto a una descripción de los motivos.

### **5.3.3 Un Prototipo de WorkFlow Social – SocialFlow**

Como hemos detallado en el Capítulo 2, el software social posee características propias y ha tenido una evolución en los últimos tiempos a instancias de lo que se denomina la Web 2.0, habiendo variados ejemplos de herramientas que lo implementan.

En torno a BPM y sus características sociales, los trabajos de investigación relacionados se encuentran en sus etapas iniciales. Las herramientas de BPM recién están incursionando en la incorporación de características sociales y no ha sido aún un campo demasiado explorado.

Los nuevos conceptos acerca del significado de un BPM con características sociales, que se focaliza en favorecer la participación de los usuarios, cobra un

sentido ampliado dado que, precisamente, un BPMS requiere la intervención constante y permanente de los “participantes” de los procesos y las herramientas que dan soporte de ejecución de procesos, no consideran aún este aspecto y mucho menos lo incluyen en el ciclo de vida de los procesos de negocios. Debido a esto, en esta tesis se presenta un prototipo de solución – SocialFlow, que incluye estas características funcionales, mostrando la posibilidad de considerar nuevos rastros de ejecución y las ventajas de enriquecer la colaboración y participación de los actores, además de contribuir a enriquecer el seguimiento y monitoreo de procesos de negocio [147].

SocialFlow es una aplicación que permite efectuar el seguimiento del flujo de procesos de negocio y sus distintas instancias, según el orden y estado de las actividades que los componen y los permisos de acceso que se otorguen a los usuarios involucrados en los procesos. Asimismo, SocialFlow permite incorporar comentarios a cada instancia de las actividades de los procesos, mantener conversaciones entre los participantes, etiquetar participantes y recibir notificaciones ante cada uno de estos eventos.

Las notificaciones y los comentarios etiquetados, permiten fijar nuevos indicadores para evaluar a la hora de analizar el proceso y su desarrollo. A modo de ejemplo, tales indicadores incluyen: el número de actores no habituales que han intervenido en un proceso, el número de tareas donde actores no habituales tuvieron mayor porcentaje de participación que los actores habituales, y el número de veces que un actor habitual requirió intervención de actores no habituales.

A diferencia con un BPMS tradicional, los rastros de las notificaciones recibidas y en qué momento fueron confirmadas, permite conocer los tiempos de demora entre cada hito, independientemente de la compleción de las actividades. Por otra parte, los comentarios etiquetados dan cuenta de la intervención de los actores independientemente de la manera en que se definieron los participantes del proceso. De este modo, es posible establecer estadísticas sobre el grado de participación de todos los involucrados y otorgar una calificación de reputación a los participantes en base a sus contribuciones.

A continuación se explica la arquitectura de la solución propuesta.

### **Tipo de solución y arquitectura**

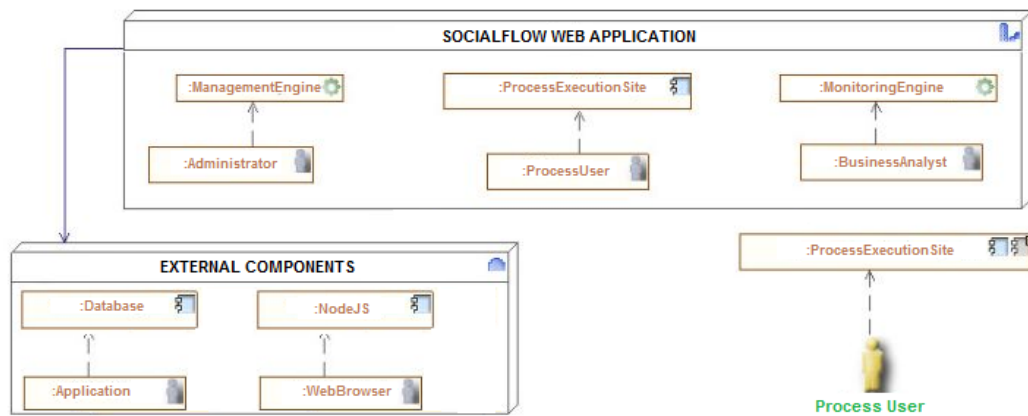
El tipo de solución propuesta es una aplicación Web, con un arquitectura de capas como se muestra en la Figura 37. Los componentes utilizan mecanismos de comunicación asincrónica para dar respuesta al esquema de eventos y notificaciones.

La arquitectura se compone de dos grandes paquetes:

**SocialFlow Web Application** representa la aplicación web que contiene el entorno de ejecución de los procesos, y los motores para la gestión y el monitoreo de los mismos (*:ProcessExecutionSite*, *:ManagementEngine* y *:MonitoringEngine*, respectivamente)

**External Components**, incluye la base de datos (*:Database*) que permite la gestión del *workflow*, implementada en MSSQL 2008 y que establece una conexión

SQL entre el DBMS y la aplicación web. Contiene también *:NodeJS*, tecnología asincrónica que permite realizar tareas de monitorización hacia la base de datos (*pooling*), y en base a los resultados obtenidos enviar notificaciones a los clientes web (*:WebBrowser*)



**Figura 37. Arquitectura y Componentes Tecnológicas de SocialFlow**

### Actividades como servicios

Las actividades de los procesos de SocialFlow son implementadas a través de servicios. Dichos servicios pueden ser desarrollados a través de formularios en cualquiera de los lenguajes presentes actualmente en la industria. Estos formularios contarán a su vez con su propio entorno de ejecución, y sus propios datos. El cliente interactúa con los formularios a través de un *iFrame* (elemento HTML que permite incrustar un documento HTML dentro de otro) de dentro de la interfaz. SocialFlow provee una API que permite que los formularios externos puedan hacer uso de operaciones que SocialFlow hace públicas como por ejemplo, acceder a las variables de procesos o completar una actividad de proceso.

A modo de ejemplo, la Figura 38 muestra un ejemplo de formulario asociado a la tarea de *Purchase Order Evaluation*, construido en SocialFlow. Los botones de *Cancel* y *Complete Activity* están asociados al contexto de la actividad e intercambian información con el formulario interno a través de variables de procesos. Asimismo, el formulario permite incluir comentarios y adjuntar archivos que quedarán asociados a esta instancia de actividad.

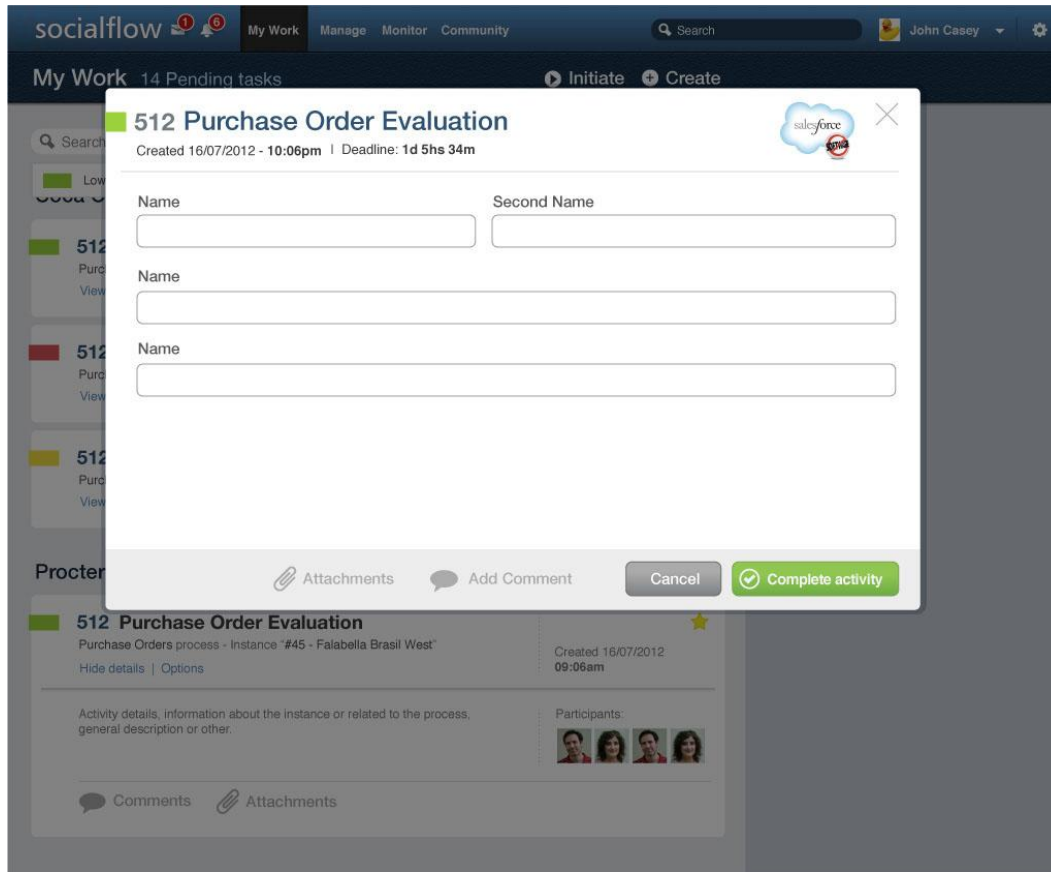
Cuando se utiliza la opción *Add Comment* se visualiza la interfaz presentada en la Figura 39. Al ingresar el comentario pueden etiquetarse a participantes del proceso al cual pertenece la actividad y también introducir palabras claves utilizando algún tipo de hash-tag.

### Notificaciones internas

Pueden ocurrir ciertos eventos en forma interna dentro de la plataforma Socialflow que requieren ser notificados a los clientes sobre su ocurrencia. *NodeJS* se encarga de hacer *pooling* en forma continua sobre la base de datos, y de esta

manera al detectar algún cambio sobre los eventos configurados, envía notificaciones a los clientes involucrados.

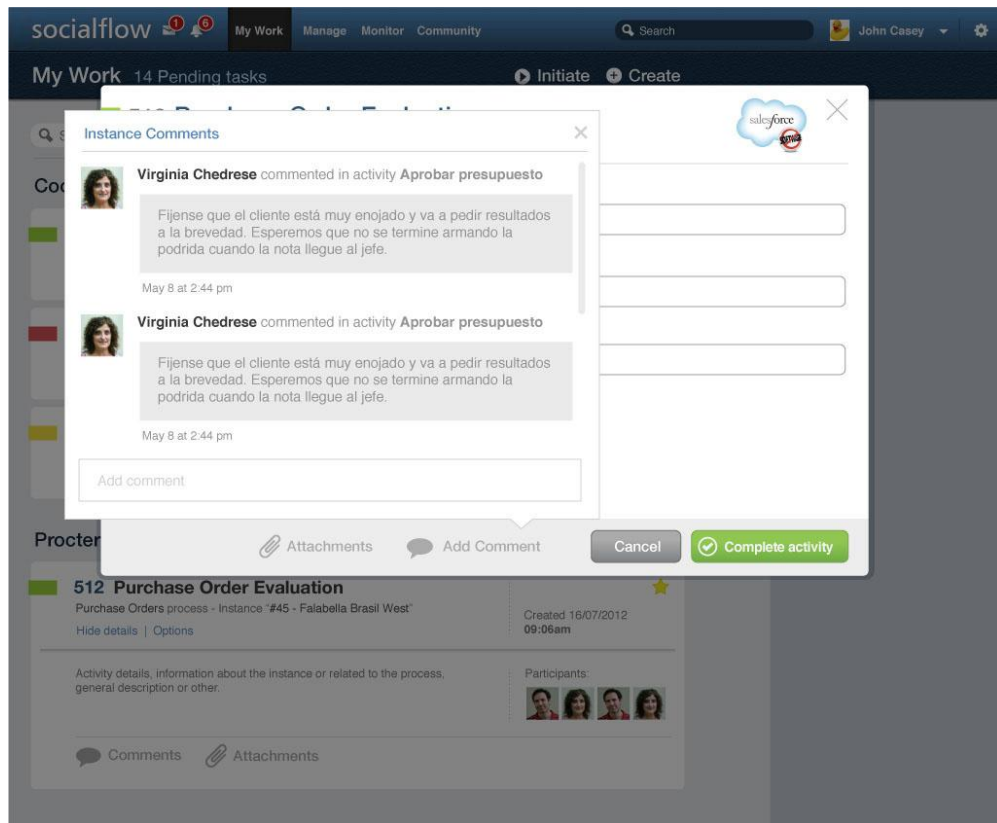
La Figura 40 muestra el mecanismo de gestión de las notificaciones implementado por SocialFlow.



The screenshot displays the SocialFlow user interface. At the top, there is a navigation bar with 'socialflow' branding, notification icons, and user information for 'John Casey'. Below this, a 'My Work' section shows '14 Pending tasks' and buttons for 'Initiate' and 'Create'. A modal window is open, titled '512 Purchase Order Evaluation', with a 'Salesforce' logo and a close button. The modal contains a form with the following fields: 'Name' and 'Second Name' (two input boxes), followed by three individual 'Name' input boxes. At the bottom of the modal, there are buttons for 'Attachments', 'Add Comment', 'Cancel', and 'Complete activity'. The background shows a list of tasks, with the selected task '512 Purchase Order Evaluation' expanded to show details like 'Purchase Orders process - Instance #45 - Falabella Brasil West', creation date '16/07/2012 09:06am', and a list of participants.

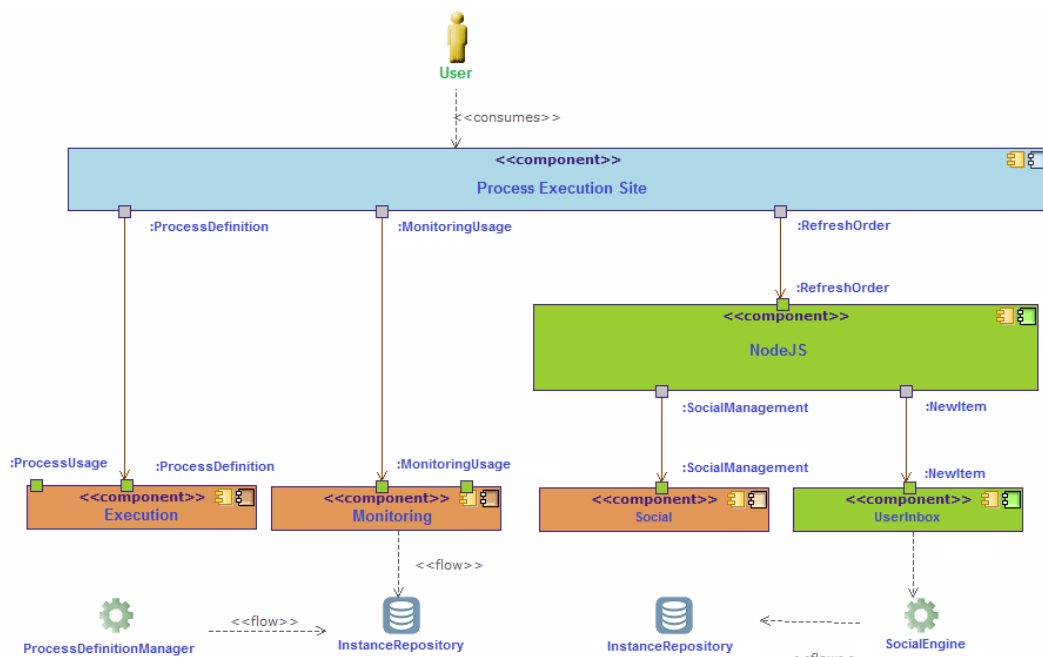
**Figura 38. Formulario Asociado a una Tarea en SocialFlow**



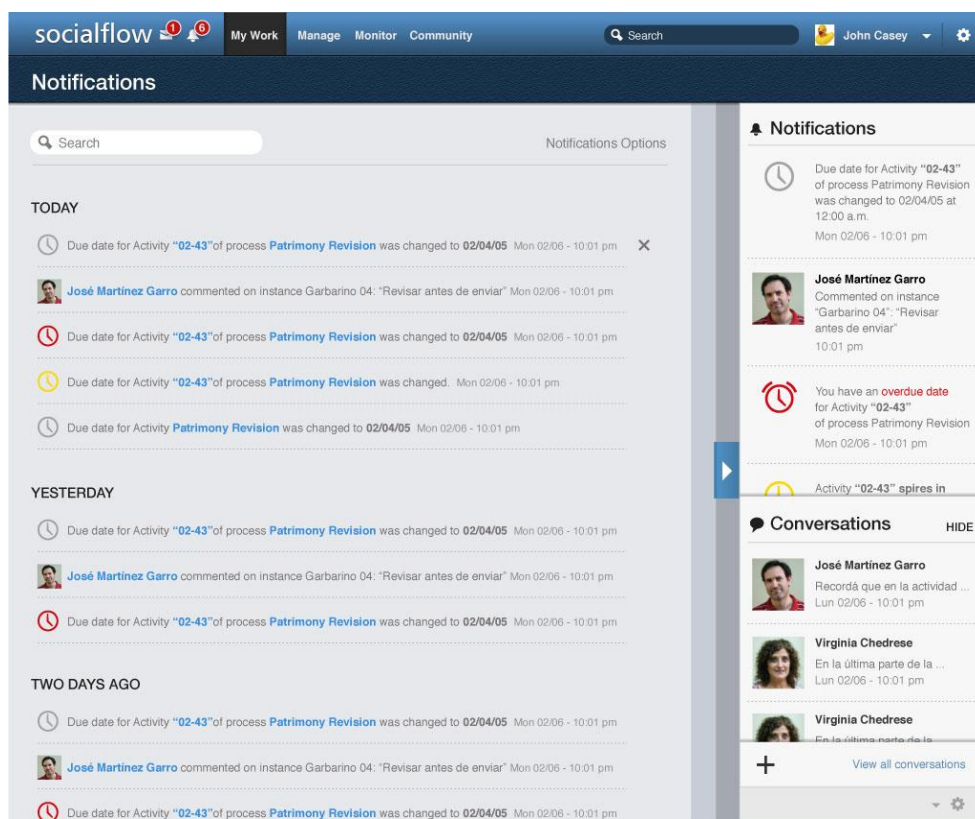


**Figura 39. Creación y Edición de Comentarios en SocialFlow**

Las notificaciones recibidas por cada usuario de SocialFlow, pueden visualizarse en forma completa y ordenadas por fecha de recepción. En caso que la actividad no se encuentre terminada, las notificaciones permiten al usuario dirigirse directamente a ver detalles de dicha actividad. La Figura 41 muestra la interfaz de visualización de notificaciones provista por SocialFlow.



**Figura 40. Mecanismo de Notificaciones en SocialFlow**



**Figura 41. Visualización de Notificaciones en SocialFlow**

### Espacio de trabajo del usuario

Como hemos mencionado, SocialFlow permite realizar el seguimiento de procesos de negocios incorporando conceptos de software social. Las notificaciones y comentarios se introducen en el espacio de trabajo del usuario bajo el mismo concepto que se utiliza en las redes sociales. Estas funcionalidades orientadas a eventos, permiten incorporar una comunicación no estructurada a la ejecución de los procesos y dejar los rastros correspondientes.

En la Figura 42 se observa una captura de pantalla del espacio de trabajo del usuario. En el panel izquierdo se encuentra la lista de ítems de trabajo en los cuales el usuario tiene que participar y que conforman las actividades de las instancias de procesos. Las mismas se agrupan según el proyecto o tipo de proceso al que corresponden (En el ejemplo: Coca Cola o Procter & Gamble). En el panel derecho, y con opción a ocultarse, se encuentran las notificaciones recibidas y un canal de conversación interna donde el usuario puede comunicarse con otros participantes de los procesos en los que se encuentra involucrado y que comparten actividades con él.

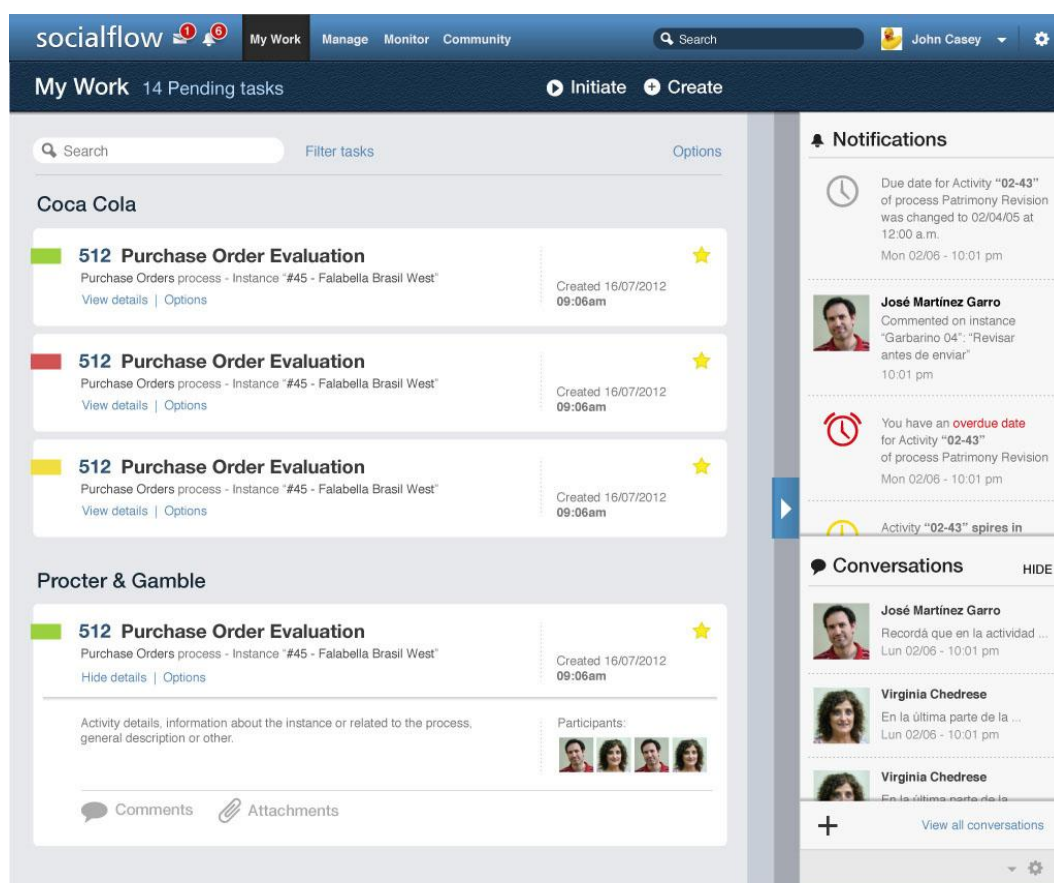


Figura 42. Espacio de Trabajo del Usuario de SocialFlow

## 5.4. CONCLUSIONES DEL CAPÍTULO

En este capítulo se presentó una propuesta para incorporar características distribuidas así como también sociales y colaborativas a la gestión por procesos de negocio.

Esta contribución está directamente vinculada a aspectos tecnológicos, como son la provisión de herramientas, principalmente, para las etapas de promulgación y evaluación del ciclo de vida de los procesos de negocio. Las herramientas encajan dentro del ciclo metodológico planteado en MISP, y presentado en el Capítulo 2.

Para alcanzar el objetivo planteado se revisaron diferentes soluciones tecnológicas para implementar procesos distribuidos y se construyó un prototipo de Workflow Social que muestra de qué manera se pueden obtener rastros de ejecución enriquecidos por la interacción de los usuarios en un entorno colaborativo, más allá de la traza clásica de un proceso de negocios.

En el capítulo siguiente se presenta un caso de estudio donde se aplica la metodología MISP mejorada, se analiza su distribución en la nube, y se revisan las ventajas de contar con características sociales para la ejecución del proceso.

## CAPÍTULO 6

### CASO DE ESTUDIO

Luego de presentar las mejoras propuestas para MISP (Capítulo 4) en torno al modelado de procesos y servicios y de analizar y plantear nuevos enfoques de implementación de procesos en torno a su ejecución y despliegue (Capítulo 5), este capítulo provee una evaluación de ambos aspectos, a través del desarrollo de un caso de estudio basado en un proceso de negocio vinculado al seguimiento y control de las tesinas de grado de las carreras de Licenciatura en Informática y Licenciatura en Sistemas de la Facultad de Informática de la Universidad Nacional de La Plata.

La evaluación se lleva a cabo en varias etapas. Primero, se plantea el problema a resolver, luego se explica cómo aplicar MISP mejorada en la etapa de modelado de procesos y servicios. A continuación se detalla el impacto de incluir enfoques distribuidos y sociales para las etapas de ejecución y despliegue del proceso de negocio. Finalmente se elaboran algunas conclusiones

#### 6.1. PLANTEO DEL PROBLEMA

El proceso de negocio que se analiza como caso de estudio realiza el seguimiento y control de las tesinas de grado de las carreras de Licenciatura en Informática y Licenciatura en Sistemas de la Facultad de Informática de la Universidad Nacional de La Plata. Al proceso lo denominamos TesinaFlow.

El proceso se inicia con la presentación de una propuesta de tesina por parte de uno o dos alumnos, que cuentan con el aval del o de los director(es). Para cada tesina presentada, luego de validar los datos académicos, se establece un Tribunal o Jurado que revisará la propuesta. Previo al establecimiento del Tribunal, se notifica a cada miembro mediante una notificación. El profesor notificado puede aceptar o rechazar la propuesta. El Tribunal es el encargado de evaluar la tesina cuando esta se entrega.

Al momento de la presentación de la propuesta, el Tribunal puede aceptarla, rechazarla o bien solicitar reformulaciones parciales o totales a la misma. En todos los casos se notifica al/los alumnos y al/los directores. Finalizado el trabajo de tesina, los alumnos presentan el informe final.

Durante el tiempo que los alumnos demoran en preparar la tesina, el Jurado puede ser ratificado o rectificado, debiendo comunicarse al/los alumno/s y al/los director/es de dicha conformación. Asimismo, se notifica al nuevo miembro seleccionado para integrar el Jurado, como se hace al momento de designar un Jurado como evaluador de la tesina. En esta instancia el nuevo profesor seleccionado puede aceptar o no su designación como miembro del Jurado.

Acordada la conformación del tribunal, el mismo evaluará el informe y determinará una fecha de exposición o bien formulará pedidos de revisión. En ambos casos se notifica al/los alumno/s y al/los director/es. Luego de acordada la fecha de exposición, se lleva a cabo la misma, el Jurado emite su opinión, redacta el acta correspondiente en el libro de actas y fija la calificación. El proceso finaliza con la graduación del alumno.

### **Áreas y actores involucrados en el problema**

El proceso TesinaFlow que se ha descrito atraviesa diversas áreas e involucra a diversos actores, siendo la identificación de los mismos, parte de las actividades que implica abordar el problema con enfoque de procesos y servicios.

Para aportar mayor claridad al planteo del problema, se describen las áreas y actores más relevantes en el proceso TesinaFlow:

La Coordinación de Tesinas es un área dentro de la institución educativa que tiene por objetivo llevar el registro y seguimiento de los alumnos que inician el desarrollo de su tesina de grado, permitiendo registrar la interacción entre los involucrados (alumnos, directores y evaluadores) e interactuar con las otras áreas a los efectos de verificar reglas y transmitir información.

El sistema SIU-Guaraní es un sistema informático que lleva adelante la gestión de los alumnos en cuanto a los datos registrales de los mismos que conforman su historia académica en el marco de un plan de estudios de una carrera. La tesina de grado representa una instancia más dentro del trayecto curricular y existen reglas específicas para poder iniciar su desarrollo.

Área de Enseñanza, involucra el sector administrativo que se ocupa de la atención a los alumnos de las diferentes carreras de la Facultad y que gestiona el flujo de expedientes, emite reportes, es usuario del sistema SIU-Guaraní en interviene en todo lo referido a notificaciones y trato con los alumnos.

## **6.2. ETAPAS 1 A 3 – PLAN, REQUISITOS Y MODELADO DEL NEGOCIO**

En esta sección se describe la aplicación de MISP para el proceso del caso de estudio y se muestra el impacto de aplicar la versión mejorada, incluyendo el modelado de procesos y servicios, la generación de componentes y el WSDL de los servicios web, según lo presentado en el Capítulo 4.

Las etapas 1 y 2 de MISP proponen plantear la organización y plan estratégico del proyecto y la identificación y especificación de requisitos con diseño participativo de procesos, respectivamente. Estas etapas cobran relevancia en proyectos organizacionales complejos donde la cantidad de procesos es grande y amerita un análisis más global. La aplicación de estas etapas fue ampliamente discutida en [1] y no constituyen el foco central de la mejora propuesta a MISP. De manera similar, la etapa 3, Modelado del Negocio, corresponde a la identificación del mapa de procesos, que también cobra sentido cuando existe una cantidad considerable de procesos a modelar dentro de la organización. Por

estos motivos, aplicamos MISP a partir de la etapa 4, Modelado de Procesos. La aplicación de MISP a partir de la etapa 4 se explica en las secciones subsiguientes.

### 6.3. ETAPA 4 - MODELADO DE PROCESOS

A partir de la descripción planteada para el proceso de negocio para el seguimiento de tesina de grado, en adelante TesinaFlow, se puede identificar claramente el flujo de trabajo y los actores intervinientes, modelando diferentes sub-procesos en notación BPMN, tal como se muestra en la Figura 43. En dicha figura se representa el flujo de trabajo existente en la definición del problema y se arriba al mismo mediante la identificación de actores y la interacción con los mismos en un diseño participativo del proceso [1].

Entre los actores principales de TesinaFlow se identifican *Alumno*, *Área Administrativa* y *Tribunal*. Por otra parte, y siempre como consecuencia del diseño participativo de procesos, se identifican las actividades que cada actor realiza y, según su grado de complejidad, se las conceptualiza como un sub-proceso. Por ejemplo, el sub-proceso propuesta de tesina, constituye el primer conjunto de actividades que el *Alumno* realiza. Si bien de la descripción del problema, esta sería una actividad simple o tarea, del Reglamento de Tesinas de Grado de la Facultad de Informática, se desprende que *Presentar Propuesta de Tesina* no representa una actividad simple sino que contiene una serie de pasos a realizar por el actor, que amerita su definición como sub-proceso (Ver Figura 44 más adelante).

Según se explicó en el Capítulo 4, el modelado y clasificación de procesos se aplica identificando distintos niveles de abstracción dentro del proceso como los subprocesos y las actividades. Para ello se instancia el metamodelo de BPMN utilizando los diferentes elementos del lenguaje.

Los actores son instancias de *pool*. El proceso es una instancia de *ProcessObject* que es una generalización de *Activity*, que a su vez se clasifica en *Task* o *Subprocess* y de *FlowObject*, que refiere *Activity*.

#### 6.3.1. Modelado de los Subprocesos

A fin de refinar el proceso TesinaFlow, se modelan cada uno de los subprocesos también en notación BPMN. En esta sección, presentamos los gráficos correspondientes a cada uno de los subprocesos donde se visualiza el modelo de interacción entre actividades. En particular, el proceso TesinaFlow se divide en los siguientes sub-procesos: 1) Propuesta de tesina, 2) Validación de datos académicos, 3) Análisis de propuesta, 4) Desarrollo la tesina, 5) Presentación de informe final, 6) Evaluación informe, 7) Coordinación fecha de exposición, y 8) Exposición de Tesina.

A continuación, se explica y se ilustra cada uno de estos subprocesos.

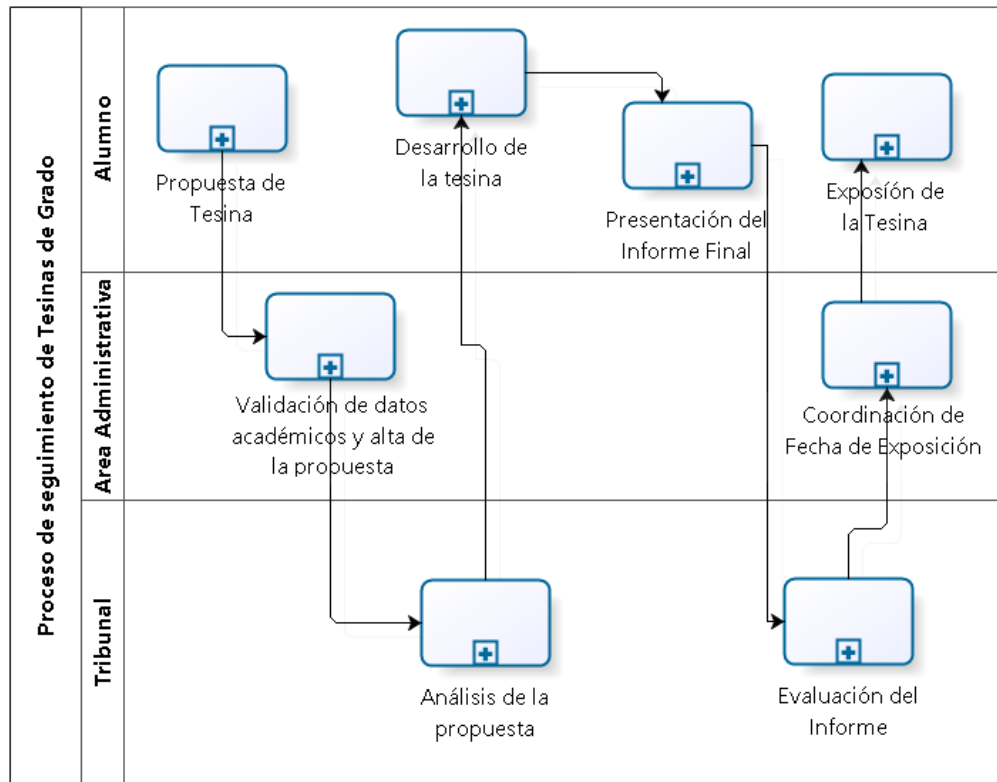


Figura 43. Subprocesos y Actores del Proceso TesisFlow

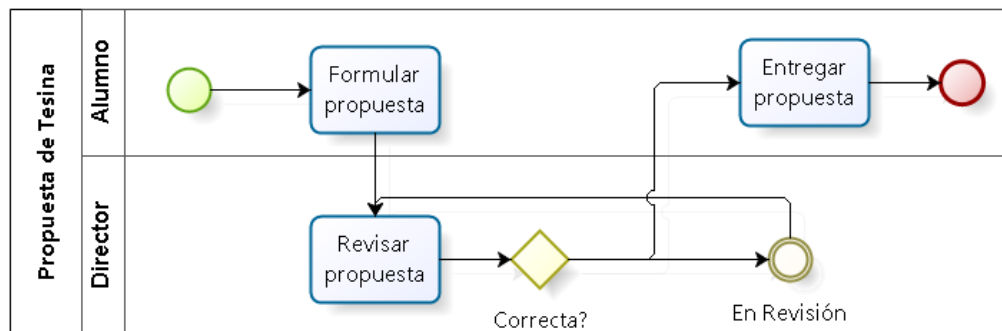


Figura 44. Subproceso Propuesta de Tesina

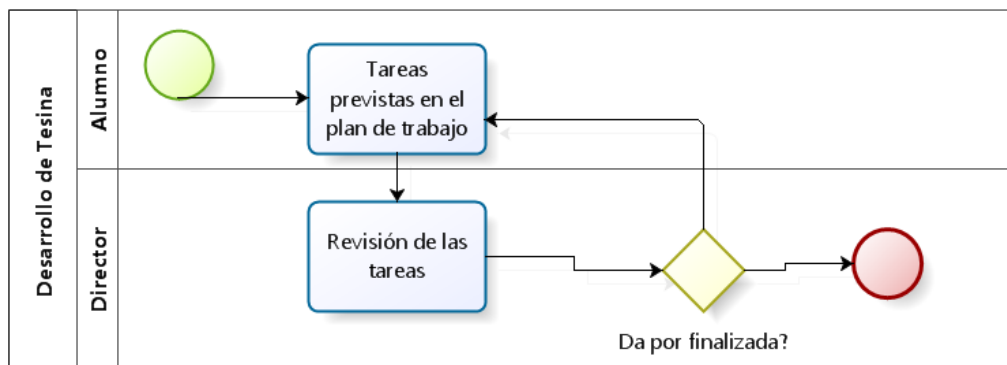
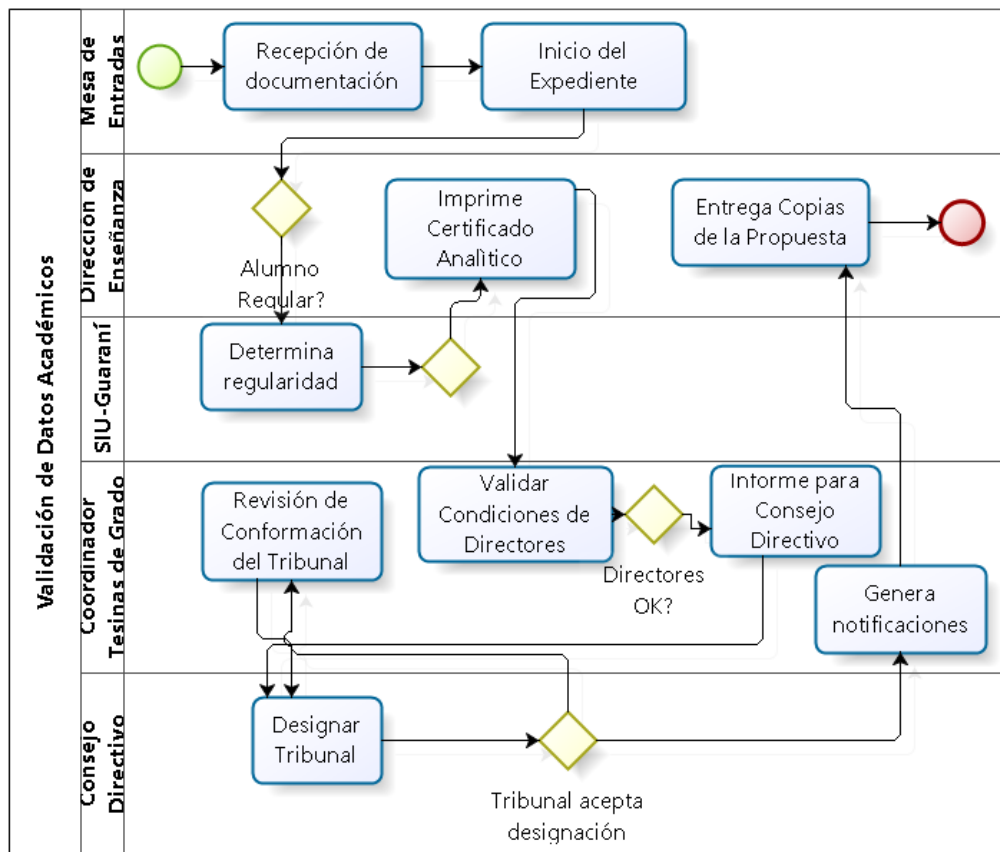
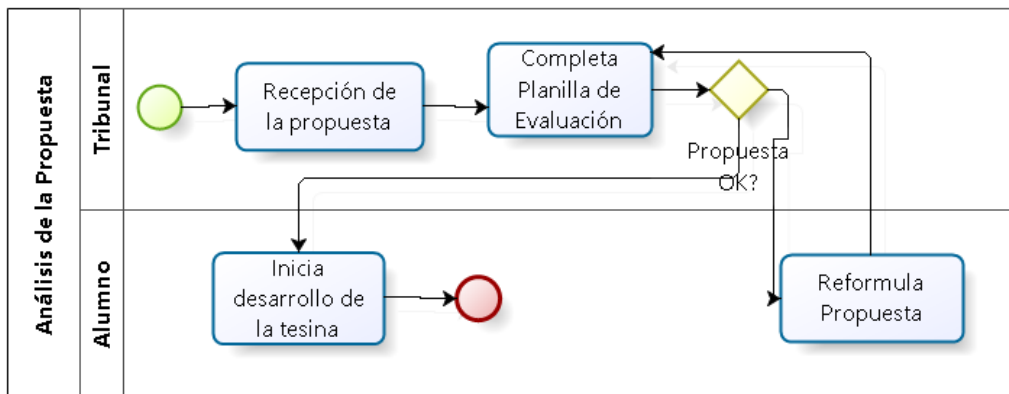


Figura 45. Subproceso Desarrollo de Tesina

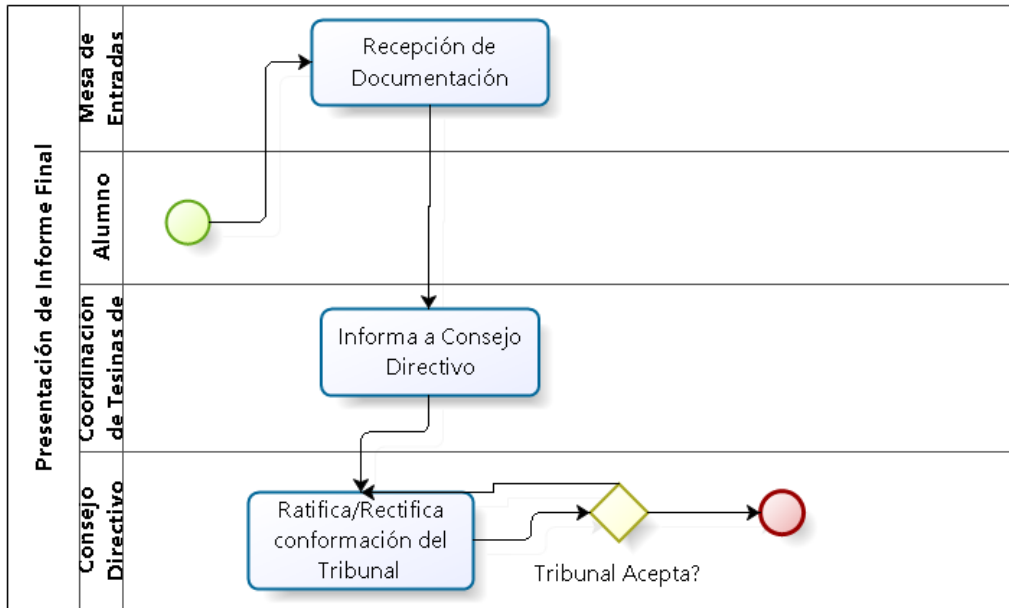


**Figura 46. Subproceso Validación de Datos Académicos**

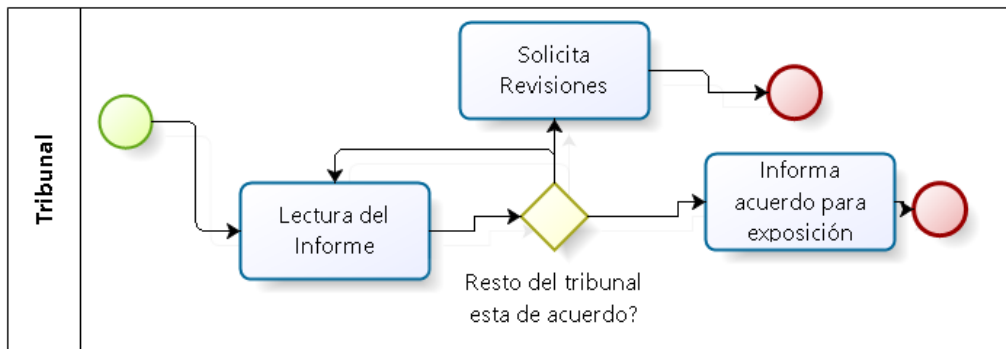


**Figura 47. Subproceso Análisis de la Propuesta**

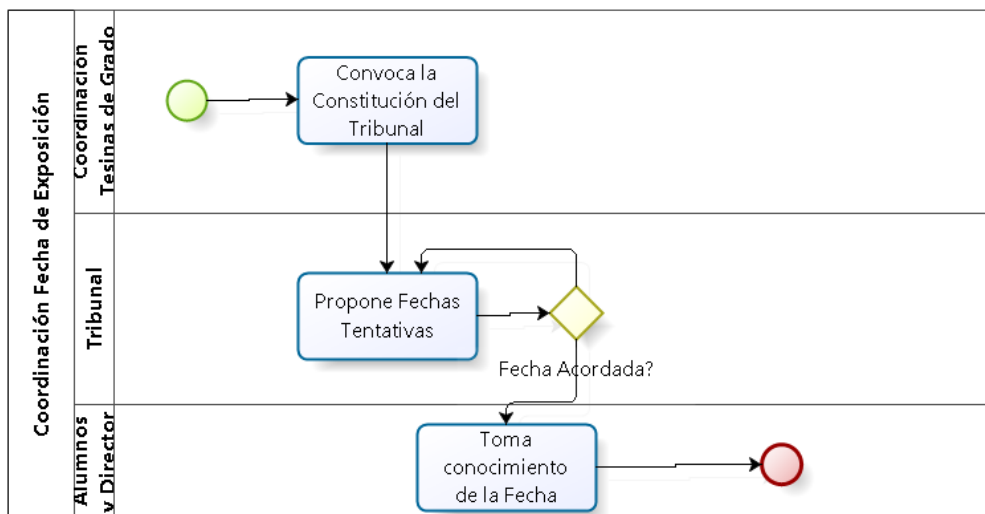




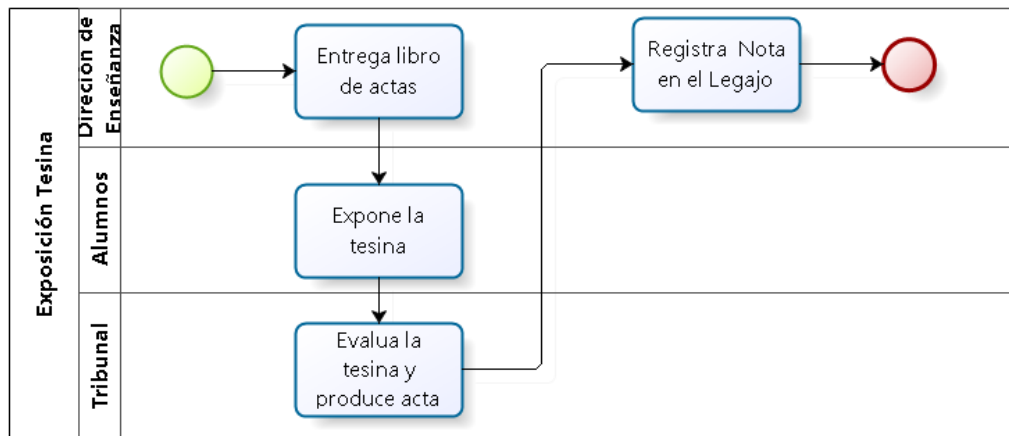
**Figura 48. Subproceso Presentación de Informe Final**



**Figura 49. Subproceso Evaluación de Informe**



**Figura 50. Subproceso Coordinación de Fecha de Exposición**



**Figura 51. Subproceso Exposición de Tesina**

Los subprocesos modelados presentan las actividades concretas y los participantes involucrados en cada una de las mismas. Este modelo representa el flujo de interacción y pone a la luz las tareas que deben llevarse a cabo, las cuales no se encuentran aún clasificadas y cuya implementación tampoco está definida. Siguiendo la metodología MISP, se realiza ahora un modelado de servicios, identificados a partir del problema y analizados en función de los actores involucrados y su interacción con los elementos del dominio.

#### 6.4. ETAPA 5 - MODELADO DE SERVICIOS (P2S).

A la luz de los subprocesos identificados y modelados en BPMN, se puede comenzar a modelar servicios asociados a cada una de las actividades tal como lo propone el metamodelo P2S y como se explicó en la Sección 4.4.3 y 4.4.4.

Luego de aplicar un análisis top-down para identificar actividades, se aplica un análisis bottom-up para obtener las componentes funcionales que responden a las actividades de negocio. La premisa para determinar una componente funcional es que contiene la información a ser administrada, las operaciones requeridas para manipular tal información y las reglas de negocio que gobiernan tal manipulación.

Este análisis bottom-up se realiza a partir de la clasificación de actividades de cada sub-proceso según la Figura 2 del Capítulo 2 y que corresponde a una taxonomía clásica que ayuda a identificar servicios que son implementados por sistemas informáticos. Dicha clasificación se muestra en la Tabla 5.

Para una revisión detallada de la etapa se elige el Subproceso de Validación de Datos Académicos a los efectos de mostrar el modelado de servicios con P2S aplicando las reglas de interacción entre metamodelos presentada en el Capítulo 4. Las actividades de este sub-proceso están detalladas en la Figura 52.

Aplicando las reglas de transformación (Tabla 4) y la metodología bottom-up para identificar y modelar servicios, se asocia el *Pool* SIU-Guaraní a un *ProcessService* en P2S, que luego se vinculará a un *Component* en SCA.

Aplicando las reglas de transformación (Tabla 4) y la metodología bottom-up para identificar y modelar servicios, se asocia el pool SIU-Guaraní a un *ProcessService* en P2S, que luego se vinculará a un *Component* en SCA.

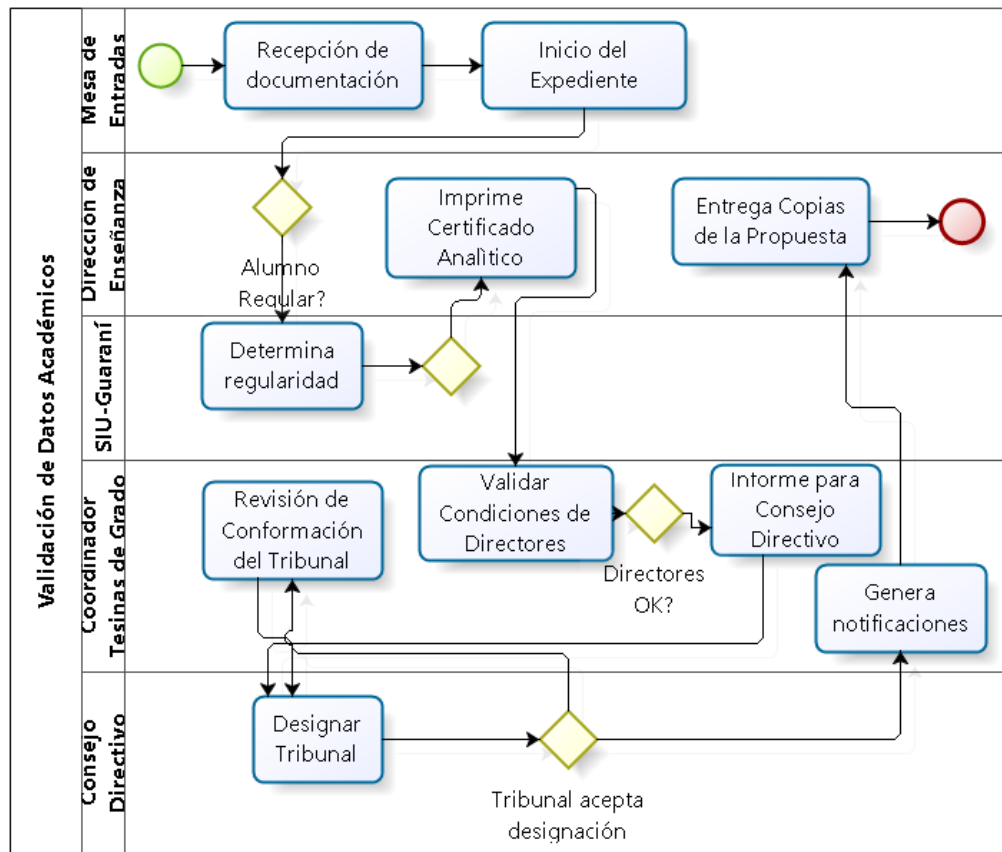
El sistema SIU-Guaraní lleva adelante la gestión de los alumnos en cuanto a los datos registrales de los mismos que conforman su historia académica en el marco de un plan de estudios de una carrera. La tesina de grado representa una instancia más dentro del trayecto curricular y existen reglas específicas para poder iniciar su desarrollo.

El *ProcessService* SIU-Guaraní constituye un servicio conceptual que implementa las capacidades de Verifica Regularidad y Verifica Condición Presentación Propuesta de Tesina, como se muestra en la Figura 53, siendo NroLegajo el contrato de uso. Este servicio conceptual se descubre realizando un análisis bottom-up de los activos de software existentes en la institución.

**Tabla 5. Clasificación de Actividades de cada Sub-Proceso**

SubProceso	Metaclase BPMN	Nombre de la Instancia	Tipo de Actividad
Propuesta de Tesina	Pool	Director	
		Alumno	
	Activity	Formular Propuesta	Manual
		Revisar Propuesta	Manual
		Entregar Propuesta	Interacción de Usuario
Análisis de la Propuesta	Pool	Alumno	
		Tribunal	
	Activity	Recepcion Propuesta	Manual
		Completar Planilla Evaluacion	Manual
		Inicia Desarrollo Tesina	Manual
		Reformular Propuesta	Manual

SubProceso	Metaclase BPMN	Nombre de la Instancia	Tipo de Actividad
Desarrollo de Tesina	Pool	Director	
		Alumno	
	Activity	Tareas Plan de Trabajo	Manual
		Revisión de Tareas	Manual
Presentación Informe Final	Pool	Consejo Directivo	
		Coordinación Tesinas	
		Alumno	
		Mesa de Entradas	
	Activity	Recepcionar Documentación	Interacción de Usuario
		Informe a Consejo Directivo	Interacción de Usuario
		Ratificar Tribunal	Manual
Evaluación de Informe	Pool	Tribunal	
	Activity	Lectura Informe	Manual
		Solicita Revisión	Manual
		Acuerdo Exposición	Manual
Coordinación Fecha Exposición	Pool	Alumno	
		Director	
		Tribunal	
			Coordinación de Tesina
Activity	Convoca Constitución Tribunal	Interacción de Usuario	
		Propone Fechas	Manual
		Toma conocimiento Fechas	Manual
Exposición tesina	Pool	Tribunal	
		Alumno	
		Dirección de Enseñanza	
	Activity	Entrega Libro de Actas	Manual
		Expone la Tesina	Manual
	Evaluar Tesina	Manual	
		Registrar Nota en Legajo	Interacción de Usuario
Validación Datos Académicos	Pool	Consejo Directivo	
		Coordinación Tesinas	
		SIU-Guarani	
		Dirección de Enseñanza	
		Mesa de Entradas	
	Activity	Recepción de documentación	Manual
		Inicio de Expediente	Interacción de Usuario
		Determina Regularidad	Sistema
		Imprime Certificado Analítico	Sistema
		Validar Condiciones de Directores	Sistema
		Informe para Consejo Directivo	Interacción de Usuario
		Revisión Conformación Tribunal	Manual
		Designar Tribunal	Manual
	Genera Notificaciones	Interacción de Usuario	
	Entrega Copia Propuestas	Manual	



**Figura 52. Subproceso Validación de Datos Académicos**

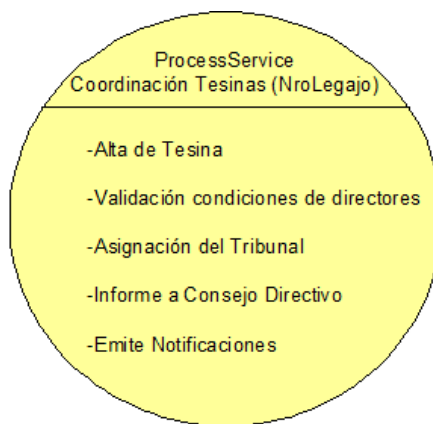


**Figura 53. ProcessService SIU-Guarani**

Aplicando las reglas de transformación según la Tabla 4 de la Sección 4.4.5, el *Pool* Coordinación Tesina se convierte en *ProcessService* en P2S y se definen sus capacidades según las actividades que se llevan a cabo en dicho pool. El análisis bottom-up descubre la existencia de un sistema de información existente dentro del área de coordinación que implementa varias de las actividades del subproceso en general y del pool *Coordinación Tesina* en particular. Esto se muestra en la figura 54 que grafica el *ProcessService* *CoordinaciónTesina*.

Las capacidades del servicio *Coordinación Tesina* incluyen:

- 1) *Alta de Tesina* - registra la tesina cuando se presenta su propuesta
- 2) *Validación Condiciones de Directores* - verifica las condiciones de profesores nominados para cubrir el rol de director previstas por el Reglamento de Tesinas de Grado
- 3) *Asignación de Tribunal* - registra los miembros del tribunal que intervendrá en la evaluación de la propuesta de tesina
- 4) *Informe a Consejo Directivo* - elabora informes al Consejo Directivo para la designación del tribunal
- 5) *Emite Notificaciones* - genera memorandos de notificación a los miembros del tribunal ante las designaciones.



**Figura 54. ProcessService Coordinación Tesinas**

Por último se conceptualiza el *ProcessService Administración y Gestión* (Figura 55), el mismo implementa las capacidades que lleva a cabo las áreas administrativas y de gestión en torno a la presentación de cada tesina de grado. Este servicio provee las siguientes capacidades:

- 1) *Imprime Certificado Analítico* - emite la historia académica del tesista
- 2) *Inicia Expediente* - conforma el trámite único que guiará el proceso de seguimiento de la tesina
- 3) *Designación del Tribunal* - representa el acto administrativo o resolución que determina los nombres del Tribunal de la tesina
- 4) *Entrega Copias de la Propuesta* - tramita la remisión de la propuesta de la tesina a cada miembro del Tribunal y
- 5) *Recepciona Documentación* - recibe y verifica la documentación presentada para completar el trámite.

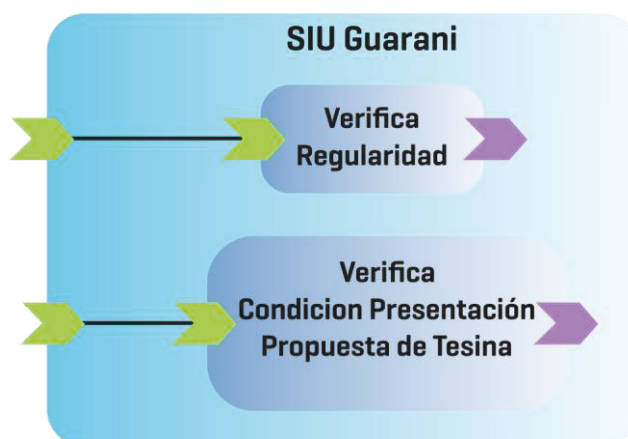


**Figura 55. ProcessService Administración y Gestión**

## 6.5. ETAPA 6 - DEFINICIÓN DE COMPONENTES.

A partir del modelo de servicios como instancias del metamodelo P2S, se alcanza la etapa 6 de la metodología que define las componentes como instancias del metamodelo SCA. Las componentes se obtienen aplicando las reglas de transformación presentadas en la Tabla 4 (Capítulo 4). Esta definición de componentes, permite vincular los servicios con las actividades de los procesos y reflejar la interacción a un nivel más cercano a la implementación. El principal objetivo de esta etapa es implementar componentes como servicios y reusar código preexistente, en la medida de lo posible, usando servicios web.

Las Figuras 56, 57 y 58 muestran instancias del metamodelo SCA que representa los componentes del ejemplo. En la figura 59 se muestra el diagrama de interacción entre componentes.



**Figura 56. Modelo de Componentes del ProcessService SIU-Guarani**

A partir de los modelos de componentes como instancias de SCA, se pueden derivar los servicios web como instancias de su metamodelo, siguiendo la integración propuesta en el Capítulo 4.

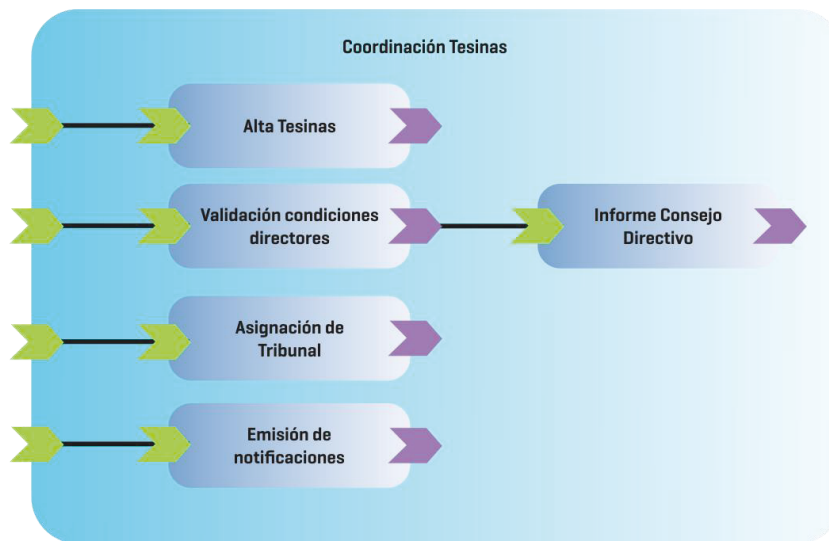


Figura 57. Componentes SCA del ProcessService Coordinación de Tesinas

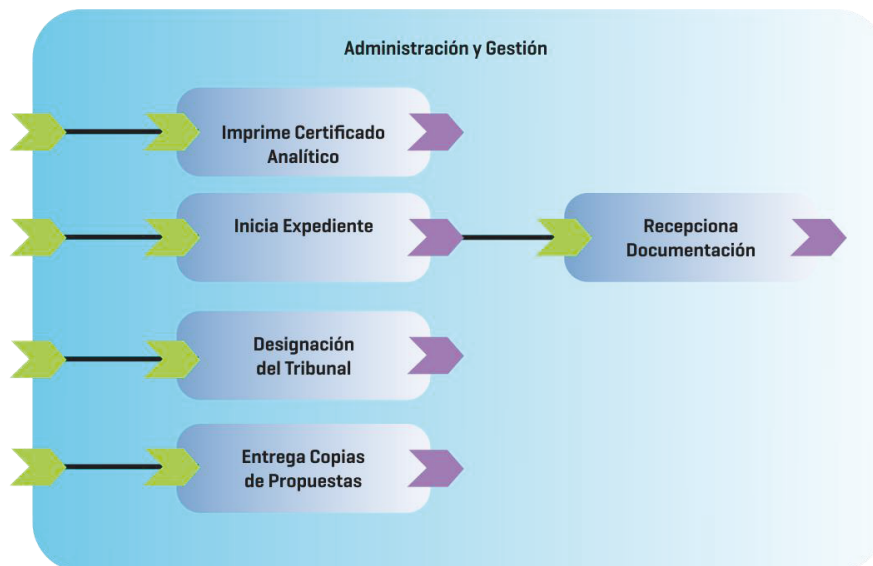


Figura 58. Componentes SCA del ProcessService Administración y Gestión

Cada *Component* de SCA de las Figuras 56, 57 y 58 representará *Definition* en WSDL. Dentro de cada *Component*, se encuentran las instancias de *ComponentService* (Ej: *VerificaRegularidad* en SIU-Guarani o *ValidaciónCondicionesDirectores* en Coordinación de Tesinas) que definen *Service* en WSDL. Cada *ComponentService* se define por sus *Operation* que se implementan con *Operation* en WSDL. Como se observa en el metamodelo simplificado SCA presentado en el Capítulo 4, la metaclass *Operation* está contenida en *BaseService* que posee una asociación con *BaseReference* que contiene *Reference*. Esta última se representa visualmente por el símbolo verde.



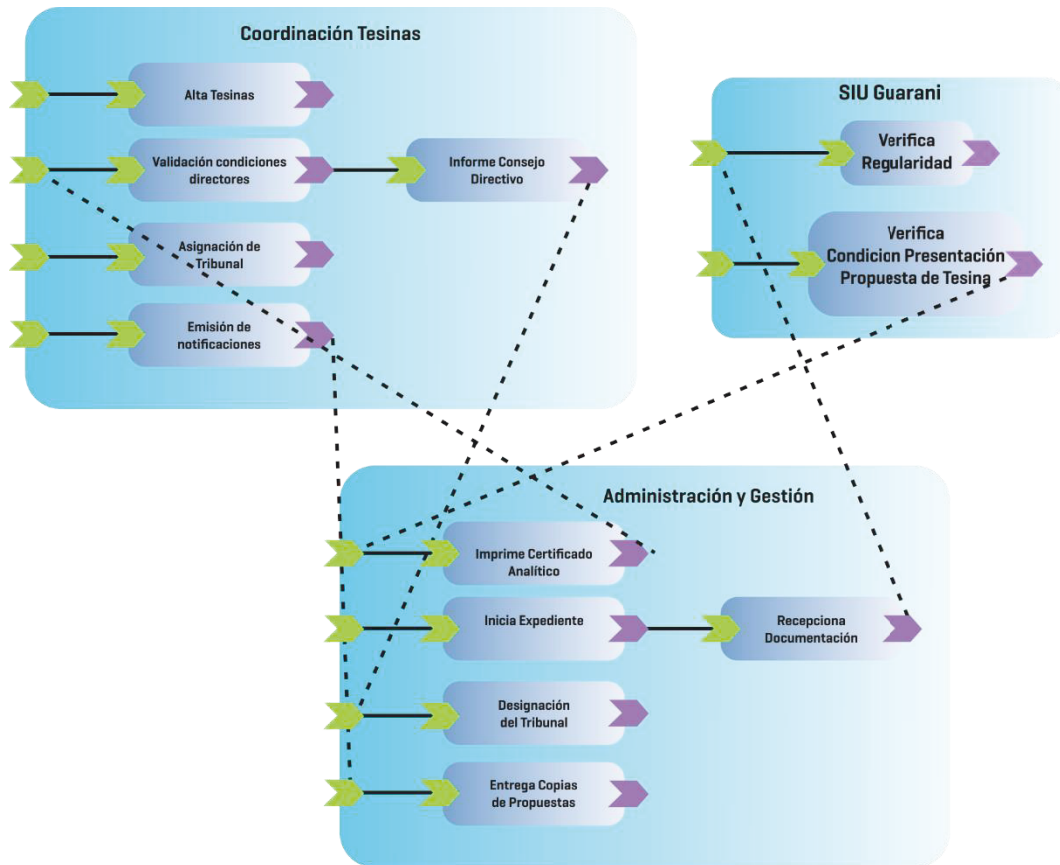


Figura 59. Modelo de Interacción de Componentes

## 6.6. INCORPORANDO ASPECTOS DE DISTRIBUCIÓN Y SOCIALES

Esta sección explica la incorporación de aspectos de distribución y aspectos sociales en las etapas de despliegue y ejecución del proceso.

### 6.6.1. Aspectos Distribuidos

En el caso de estudio elegido, en la Sección 6.3.1 se propuso una división en sub-procesos surgida de las etapas de modelado de procesos y servicios. Si bien esta descomposición emerge de las etapas de análisis y diseño, cuando se alcanza la etapa de implementación es oportuno revisar cuales de dichos subprocesos poseen actividades que requieren acceso a información propia que podría requerir cierto acceso privado así como proximidad en el tráfico. Asimismo, es importante revisar cómo se sincronizan los rastros SIU de ejecución de estos sub-procesos con el proceso completo y de qué manera se logra un monitoreo distribuido, tal como se plantea en la Sección 5.2.

En el caso de estudio del proceso *TesinaFlow*, se observa que los subprocesos que involucran al alumno y al Tribunal resultan propicios para ser desplegados en la nube a fin de facilitar su acceso a los involucrados por fuera de la infraestructura propia de la Facultad de Informática, utilizando un motor de procesos y una base de datos propia para registrar la ejecución.

Por su parte, los sub-procesos que interactúan con personal del área administrativa serán desplegados de manera embebida utilizando un motor de procesos cercano a la base de datos corporativa que debe ser accedida por los servicios que implementan las actividades de los procesos.

**Tabla 6. Distribución de Procesos de TesinaFlow**

TESINAFLOW		
SUBPROCESO/DESPLIEGUE	EN LA NUBE	EMBEBIDO
Propuesta de Tesina		
Validación de Datos Académicos		
Desarrollo de Tesina		
Presentación de Informe Final		
Exposición Tesina		
Coordinación Fecha de Exposición		
Análisis de Propuesta		
Evaluación de Informe		

Este modelo de descomposición que se presenta en la Figura 60 cuenta con las siguientes ventajas:

Los actores *Alumno* y *Tribunal* pueden ejecutar sus actividades en la nube contando con un servicio accesible vía Internet sin necesidad de habilitar el acceso a la red privada de la Facultad. A su vez, los rastros de ejecución de sus actividades pueden ser notificados cuando se alcance un hito significativo para el proceso global, evitando el registro innecesario de información y permitiendo a su vez, contar con indicadores propios.

El *Área Administrativa* cuenta con un entorno de ejecución para sus sub-procesos que se encuentra embebido dentro de la infraestructura informática de la Facultad, contando con acceso a la base de datos operativa de la Gestión de Alumnos, que exige cierta privacidad en su acceso debido a la información confidencial que contiene. Además, el tráfico de datos existente entre las actividades del sub-proceso *Validación de Datos Académicos* y los servicios que implementan las tareas de esta actividad es intenso, por lo que resulta oportuno contar con el motor de procesos y el bus de servicios ubicado dentro de la misma red, que además cuenta con buena velocidad de transmisión.

Este modelo de distribución favorece la ejecución de los subprocesos en los contextos adecuados, mejorando el rendimiento en cuanto a acceso a la información, protegiendo los datos sensibles de accesos no autorizados y externalizando los procesos útiles para su propia autogestión como, por ejemplo, el subproceso *Evaluación del Informe* o *Desarrollo de la Tesina*, que poseen características no estructuradas pero cuyos flujos de trabajo podría resultar útil de estandarizar.

El monitoreo centralizado de las instancias distribuidas, obtenido a partir de la ejecución de conectores desde un BPMS como se describe en el Capítulo 5, permite contar con una visión global de la ejecución del proceso TesisFlow y conocer los detalles de cada una de las instancias que han sido ejecutadas desde el conector, siendo este último elemento el responsable de brindar dicho detalle.

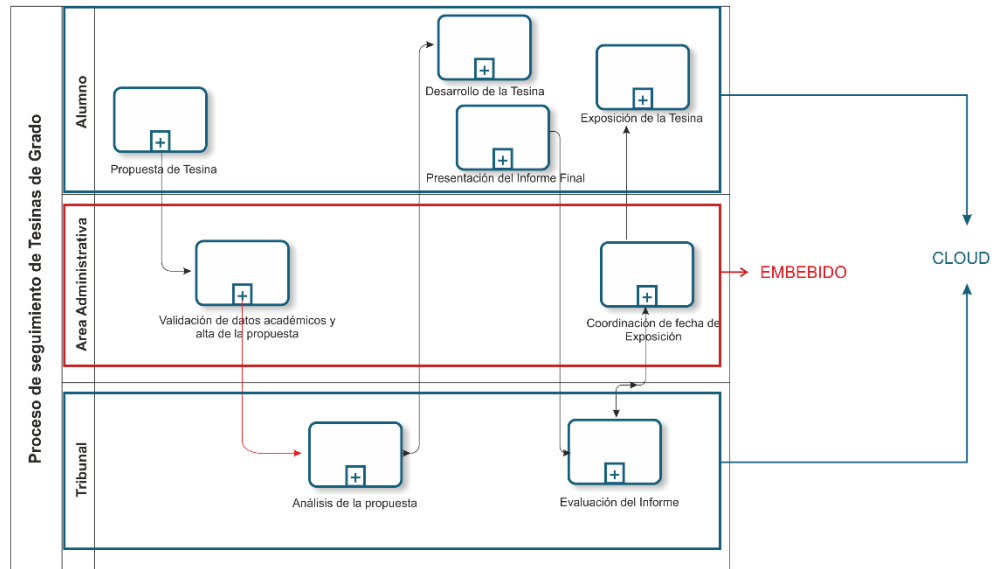


Figura 60. Descomposición y Distribución de TesisFlow

En la figura 61 se presenta de manera esquemática la asignación de cada subprocesso a cada componente de la arquitectura propuesta.

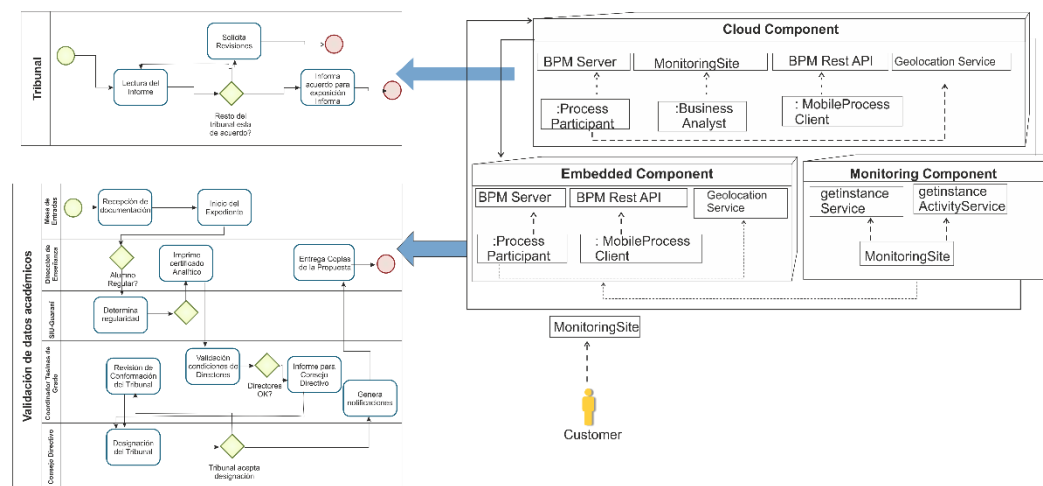


Figura 61. Subprocesos de TesisFlow en el Modelo de Componentes.

### 6.6.2. Aspectos Sociales

A fin de considerar la inclusión de aspectos sociales en las etapas de despliegue y de ejecución del proceso de negocios se debe tener en cuenta: 1) los subprocessos identificados en la Sección 6.3.1, 2) la documentación generada como

producto de la etapa 2 - "Identificación y Especificación de Requisitos con Enfoque de Procesos", y 3) el proceso de interacción de los subprocesos producido en la etapa 4 - "Modelado de Procesos".

En nuestro caso de estudio, se observa que Desarrollo de Tesina y Evaluación del Informe son claramente dos procesos complejos debido a la dificultad de modelar la diversidad de actividades que implican. Se puede afirmar que ambos son casos típicos de procesos no estructurados o dinámicos donde las actividades y los flujos varían en función de los actores que desarrollen las actividades. En este escenario, resulta oportuno aplicar técnicas de socialización de procesos que permitan dejar rastros de las interacciones a partir de comentarios y archivos adjuntos.

La interacción de los alumnos con sus directores durante Desarrollo de Tesina, así como la que se produce entre los miembros del Tribunal durante Evaluación del Informe, se ve enriquecida mediante la edición de comentarios etiquetados, parte de los aspectos sociales de los procesos de negocio que propone SocialFlow. A su vez, cada vez que se incluye una etiqueta, provoca la notificación correspondiente a los actores relacionados con la actividad, quedando el contenido de las etiquetas documentado dentro del proceso y registrado de manera automática. Estos rastros permiten realizar un seguimiento propio para cada instancia del subproceso otorgando cierta flexibilidad que no admite la definición del proceso estática tradicional.

Por otra parte, la socialización del proceso con el enfoque propuesto por SocialFlow permite incorporar actores no habituales – actores que pertenecen a la red de actores de los procesos pero que no necesariamente son participantes directos del mismo - como por ejemplo, la inclusión del alumno como actor no habitual del subproceso Evaluación del Informe, cuyos participantes principales son los miembros del Tribunal. La inclusión mediante etiquetas de estos actores no habituales, permiten notificarlos en forma automática.

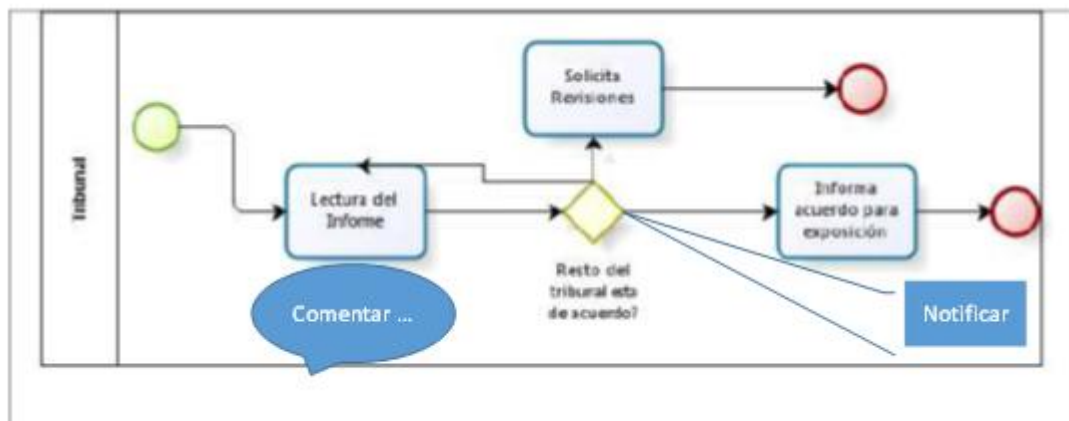
Las notificaciones y los comentarios etiquetados, permite fijar nuevos indicadores a evaluar a la hora de analizar el proceso y su desarrollo. Tales indicadores pueden ser, por ejemplo, cuántos actores no habituales de un proceso han intervenido, en cuantas tareas tuvieron mayor porcentaje de participación los actores no habituales, cuantas veces un actor habitual requirió intervención de actores no habituales, etc. Adicionalmente, los rastros de las notificaciones recibidas y en qué momento fueron confirmadas, permiten conocer los tiempos de demora entre cada hito, independientemente de la compleción de las actividades, como sucede en un BPMS tradicional. Finalmente, los comentarios etiquetados dan cuenta de la intervención de los involucrados independientemente de la manera en que se definieron los participantes del proceso. En este sentido, se pueden establecer estadísticas sobre el grado de participación de todos los involucrados y otorgar una calificación de reputación a los participantes.

A fin de ilustrar los conceptos descriptos anteriormente, utilizando el prototipo SocialFlow, se desplegaron los subprocesos Desarrollo de Tesina y

Evaluación de Informe y se generaron 10 instancias correspondientes a 10 tesis de grado en cada uno de los subprocesos.

En el caso del subproceso *Evaluación de Informe* (Figura 62), las etiquetas provistas por los miembros del Tribunal en los comentarios de la actividad Lectura del Informe y que fueron captadas por el motor de SocialFlow como parte de la ejecución del proceso, permiten determinar que en 80% de los casos, la interacción entre los miembros del tribunal requiere al menos 3 ciclos de revisión para arribar a un resultado. Asimismo, las etiquetas permiten conocer el grado de participación de cada miembro del tribunal en cada interacción en función de la cantidad de comentarios incluidos en la revisión.

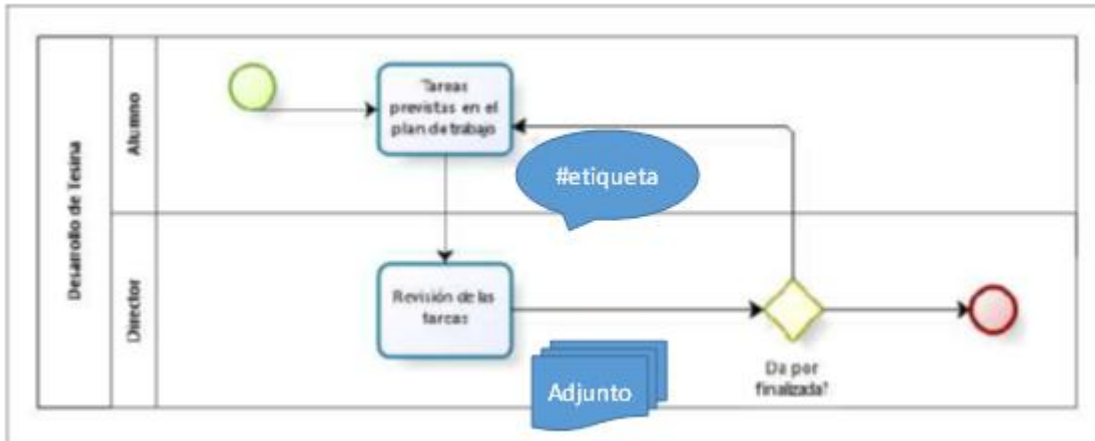
En el caso del subproceso *Desarrollo de Tesina* (Figura 63), las etiquetas y comentarios registrados tanto por los tesistas como por los directores en la "Tareas previstas en el plan de trabajo" fueron captadas por el motor SocialFlow como parte de la ejecución del proceso y permiten realizar un ranking de las subtarefas más frecuentes, identificando ciertos "hashtags". De las instancias ejecutadas se obtuvo como resultado que el 50% de las tesis tienen un grado de interacción entre directores y tesistas del orden de una vez cada 2 meses, siendo el 70% de los casos interacciones que solo involucran el intercambio de un archivo adjunto.



**Figura 62. Subproceso Evaluación del Informe-Comentarios y Notificaciones**

## 6.7. COMPARACIÓN Y BENEFICIOS DE LA SOLUCIÓN

En esta sección se presentan los beneficios obtenidos en la resolución de este caso de estudio al aplicar MISP mejorada y las actualizaciones tecnológicas incorporadas a la misma.



**Figura 63. Subproceso Desarrollo de Tesina con Etiquetas y Adjuntos**

El seguimiento de las tesinas de grado de la Facultad de Informática para las carreras de Licenciatura en Sistemas y Licenciatura en Informática representa un circuito que no se encuentra incluido en el sistema de gestión de alumnos del sistema SIU-Guaraní y que requiere consumir varios de los servicios brindados por el mismo. Además es un proceso de larga duración dado que consume entre 4 y 12 meses para su finalización. Por otra parte, muchas de las actividades previstas revisten un carácter no estructurado que resulta difícil de representar con los modelos de procesos tradicionales.

En este sentido los beneficios obtenidos en esta solución se encuentran vinculados, por un lado, a la aplicación de un enfoque de procesos clásico, mejorado con la conceptualización de servicios y por otro, la propuesta de distribución de procesos y su socialización, aportan mayor flexibilidad en su ejecución y mejor monitoreo al incluir rastros de ejecución enriquecidos.

Entre los aspectos más relevantes de esta mejora podemos detallar:

*Enfoque de procesos* – La solución de problemas con enfoque de procesos de negocio permite explicitar circuitos que de otro modo quedan ocultos en manuales de procedimiento escritos, en el mejor de los casos o incluso suele estar implícito en el quehacer de los actores que lo llevan a la práctica. A su vez permite automatizar y sistematizar tareas, identificar actores e involucrados delimitando responsabilidades y además deja rastros de la ejecución permitiendo conocer el estado de cada paso, las demoras existentes y donde se producen, logrando de este modo cerrar un ciclo de mejora continua. Todos estos aspectos no podían abordarse con la solución actual de TesisFlow.

*Integración de servicios* – El enfoque de procesos tradicional, sin la conceptualización e integración de los servicios que realizan las actividades también resultaría insuficiente debido a que no permitiría reusar activos de software existentes ni atravesar los sectores de la organización. La integración de servicios permite reutilizar servicios automáticos existentes, sistematizar servicios manuales y además explicita y especifica las interfaces con otros

sistemas informáticos. Esto fue obtenido aplicando P2S para modelar servicios a partir de un modelo de procesos de negocio desarrollado con BPMN.

*Monitoreo de actividades* – El modelado de procesos y su transformación en versiones ejecutables a través de un BPMS permite monitorear el desarrollo del proceso y conocer detalles de ejecución de sus actividades y los actores que participan que colaboran en su seguimiento y favorecen los análisis posteriores sobre dicha ejecución para identificar anomalías y proponer mejoras.

*Distribución de procesos* – La ejecución distribuida es un concepto que se aplica en diversos escenarios, siendo la ejecución distribuida de procesos uno de ellos. Bajo este modelo, es posible acercar la solución a cada actor y tener un mejor aprovechamiento de los recursos, además de exponer solamente los procesos cuyas características de tráfico de información así lo requieren. En el caso que planteamos, los procesos vinculados con los alumnos y sus directores son aptos para ser ejecutados en forma paralela con los de índole netamente administrativa, que involucra uso de servicios internos con acceso a los sistemas informáticos de la institución.

*Socialización de procesos* – La socialización de proceso permite trabajar en forma colaborativa y alterar la estructura clásica de los procesos tradicionales, adecuándolos a cada escenario. Además, deja rastros enriquecidos que representan de una mejor manera la actividad de los actores.

En la Tabla 7, se detallan los beneficios enunciados y su representación en la solución existente en comparación con la solución propuesta a TesisFlow.

**Tabla 7. Comparación de Funcionalidades**

FUNCIONALIDAD/SOLUCIÓN	SEGUIMIENTO DE TESIS DE GRADO	TESISFLOW
Registro de datos en sistemas informáticos	Si	Si
Representación conceptual del circuito	En un reglamento	En un modelo ejecutable
Monitoreo del circuito	Por seguimiento del expediente	Por ejecución de actividades
Integración con servicios	Manual	Automática
Representación de procesos no estructurados	En forma externa entre los involucrados	Comentarios, notificaciones y adjuntos
Capacidad de ejecutar procesos online	No es posible	Procesos distribuidos

## 6.8. CONCLUSIONES DEL CAPÍTULO

En este capítulo presentamos un caso de estudio utilizado para validar la propuesta presentada en esta tesis. El caso presenta un ejemplo de proceso de negocio que se desarrolla en la Facultad de Informática para el seguimiento de las tesinas de grado de los alumnos de las carreras de Licenciatura en Informática y Licenciatura en Sistemas.

La metodología MISP plantea un conjunto de etapas que cubre todo el ciclo de vida de los procesos de negocios. En las etapas de modelado de procesos y servicios, se presenta la mejora de MISP para la identificación de servicios a partir del modelado conceptual de los mismos utilizando el metamodelo P2S y sus reglas de integración con los metamodelos de BPMN y SCA, permitiendo obtener un modelo de implementación con servicios web integrando el metamodelo SCA con el metamodelo WSDL. Esta mejora ha sido validada en trabajos previos, publicados durante el desarrollo de esta tesis (Ver Capítulo 7).

Las siguientes etapas de la metodología que abarcan la promulgación de los procesos y su posterior monitoreo, involucran tecnologías y herramientas que soportan la ejecución y despliegue de dichos procesos en el marco de BPMSs. En este trabajo se proponen prototipos que permiten incorporar en dichas herramientas, capacidades para socializar procesos y para distribución de los mismos en entornos no locales a la luz de mejorar la etapa de monitoreo de procesos con rastros enriquecidos por interacciones de índole colaborativa y proveyendo un monitoreo centralizado a partir de rastros distribuidos en entornos en la nube y embebidos.

Las mejoras de modelado a través de metamodelos permiten expresar los servicios a partir de actividades de un proceso, obteniendo los servicios desde una visión bottom-up realizada desde activos de software existentes y también desde las actividades manuales que realizan los participantes de los procesos. Adicionalmente, las propuestas tecnológicas existentes que habilitan incluir capacidades distribuidas y colaborativas, plantean un enfoque que no se encuentra considerado en las herramientas actuales de manera integrada y que incluyen estrategias necesarias a la hora de dotar a los procesos de negocio de la flexibilidad que refleja la participación de actores no habituales y la ejecución en entornos externos a las organizaciones, donde la distribución de procesos requiere una ejecución no local pero donde el monitoreo como un proceso único sigue siendo necesario para cerrar el ciclo de mejora continua.



## CAPÍTULO 7

### CONCLUSIONES

En esta tesis se propusieron mejoras a la Metodología Integradora de Servicios y Procesos (MISP), proveyendo una nueva visión de los procesos y los servicios a la luz de recientes avances tecnológicos y buscando reducir la brecha entre los analistas de negocio y los expertos de tecnología. Luego de presentar los elementos conceptuales que sustentan esta propuesta (Capítulo 2), se realizó un análisis exhaustivo y detallado del estado del arte en torno a las diferentes propuestas metodológicas para la integración de procesos y servicios y las alternativas tecnológicas actuales para el despliegue y ejecución de procesos de negocio (Capítulo 3). A continuación, se propusieron las mejoras introducidas a MISP en las etapas de modelado de procesos y servicios (Capítulo 4), y las actualizaciones tecnológicas desarrolladas en esta tesis para el monitoreo de procesos de negocio que se despliegan y ejecutan dentro de un motor de procesos, principalmente teniendo en cuenta la distribución de procesos y la consideración de aspectos sociales (Capítulo 5). Las mejoras metodológicas como las actualizaciones tecnológicas se validaron mediante un caso de estudio (Capítulo 6). Finalmente, este Capítulo resume los resultados obtenidos (Sección 7.1), sintetiza las contribuciones (Sección 7.2) y se discuten posibles trabajos futuros (Sección 7.3).

#### 7.1. RESULTADOS OBTENIDOS

La tesis propone revisar y mejorar la Metodología Integradora de Servicios y Procesos (MISP) con el objetivo de perfeccionar aspectos que hacen al modelado de procesos y servicios y su interacción así como también de incluir actualizaciones tecnológicas disponibles en las herramientas para la gestión de procesos de negocios que se utilizan para el despliegue, ejecución y monitoreo.

En relación con la mejora a la interacción de procesos y servicios en MISP, se han revisado las etapas de modelado de procesos, modelado de servicios y definición de componentes. Esta mejora contribuye a achicar la brecha conceptual entre los procesos -vistos como elementos que modelan la realidad en un enfoque transversal y abarcativo de toda una organización, y los servicios - vistos como unidades funcionales verticales que implementan las distintas actividades que se llevan a cabo para ejecutar los procesos de negocio. En este contexto, se aporta un lenguaje de descripción de servicios (P2S) que conforma un modelo de servicios, visto como una instancia del metamodelo que representa dicho lenguaje. Asimismo, se plantean reglas de integración entre P2S y BPMN – como notación estándar para el modelado de procesos – y SCA – como enfoque estándar para crear, describir e integrar componentes.

En lo relativo a las actualizaciones tecnológicas que se propone incorporar a las herramientas de gestión de procesos de negocio, se han revisado la incorporación de características distribuidas y sociales dentro de un motor de procesos. En particular: 1) se propuso un mecanismo de monitoreo distribuido de procesos cuando estos se encuentran desplegados y se ejecutan en una arquitectura de aplicaciones en la nube, 2) se analizaron y se incluyeron aspectos sociales en la ejecución de los procesos, los rastros que estos aspectos permiten reunir sobre el proceso, y su manera de considerarlos en la fase de mejora continua, Estas características sociales se implementaron en un prototipo de *workflow* social – SocialFlow.

La propuesta se ha validado mediante un caso de estudio referido a un proceso de negocio existente en la Facultad de Informática, para el seguimiento y control de las tesinas de grado de las carreras Licenciatura en Sistemas y Licenciatura en Informática.

En particular, en el Capítulo 1 se presenta una introducción a los procesos de negocio y su gestión como un enfoque cada vez más adoptado en las organizaciones actuales, describiendo las características clásicas para comprender la problemática actual en torno a los procesos sociales y colaborativos y también en torno a la importancia de los servicios y su modelado, como medio de interacción entre los procesos y las componentes de software. En este capítulo también se enuncia el problema a resolver en esta tesis y se muestra el enfoque de solución.

En el Capítulo 2 se desarrollan las nociones fundamentales de los conceptos vinculados a este trabajo como lo son: 1) BPM (Business Process Management) - como paradigma para la representación explícita de los procesos de negocio con sus actividades y la ejecución de restricciones entre ellas; 2) MDD (Model Driven Development) - como una manera de construcción software en enfoque guiado por modelos; 3) *Computación en la Nube* - como un nuevo modelo de computación y de servicio para desarrollo y despliegue de aplicaciones - y 4) Software Social - como estrategia para gestionar el conocimiento dentro de las organizaciones y mejorar la interacción de los actores. Luego se analiza la vinculación existente entre BPM y computación en la nube en cuanto a considerar aspectos de globalización y distribución de los procesos de negocio, y también entre BPM y el software social en cuanto a la incorporación de sus rastros a nivel operativo durante la ejecución de los procesos de negocio.

En el Capítulo 3, se revisan trabajos relacionados y estado del arte agrupando dicha revisión en: 1) lenguajes de modelado y workflows, 2) el desarrollo dirigido por modelos y su aplicación para la gestión por procesos, 3) las propuestas actuales en torno a aplicación de computación en la nube y BPM y 4) las alternativas existentes en los últimos dos años en torno a lo que se denomina BMP social. Por otra parte se dedican dos secciones a analizar y herramientas por otro, que dan soporte al ciclo de vida de los procesos, su madurez y grado de avance.

En el Capítulo 4 se presenta la primera parte del aporte de la tesis que consiste en la mejora a MISP para la interacción entre el modelado de procesos y servicios sustentando por metamodelos. Se presenta P2S, un metamodelo propio para modelar servicios y su prototipo de implementación como plug-in de Eclipse y su integración con los metamodelos de BPMN y SCA. Otra contribución presentada en este capítulo es la comparación de propuestas existentes en la bibliografía en torno, específicamente, a la integración de procesos y servicios. Se ilustra esta parte de la propuesta con un ejemplo sencillo.

En el Capítulo 5 se aborda el segundo aporte de la tesis en cuanto a aplicar actualizaciones tecnológicas sobre las etapas de despliegue, ejecución y monitoreo de procesos, incorporando características sociales a un prototipo de *workflow* (SocialFlow) y su contribución en el monitoreo de procesos a la luz de los nuevos rastros de ejecución. Además se presenta una arquitectura para llevar a cabo el monitoreo de procesos distribuidos utilizando herramientas open-source.

Finalmente, en el Capítulo 6 se desarrolla un caso de estudio aplicando las mejoras metodológicas y las actualizaciones tecnológicas a un proceso de negocio de una institución académica.

## 7.2. CONTRIBUCIONES

Esta tesis ha realizado las siguientes contribuciones:

Aporta una mejora sustancial a una metodología propia (MISP) definiendo un lenguaje para modelado y conceptualización de servicios, en el contexto de un ciclo de vida de procesos de negocio articulado con el ciclo de vida del software. Este lenguaje de modelado se integra con BPMN y SCA logrando una interacción entre los procesos, los servicios y las componentes de software, siempre en el contexto de una visión global de una organización que aborda su gestión por procesos de negocio

Incorpora avances tecnológicos actuales en torno a nuevos modelos de computación y paradigmas de socialización del software, y su impacto en la gestión de procesos. Estas ideas se prueban mediante la implementación de prototipos de software

Brinda un exhaustivo estudio de conceptos en cuanto a metodologías y tecnologías, su impacto y modo de aplicación en BPMS, vistos como las herramientas que permiten poner en ejecución los procesos de negocios

Aporta una pormenorizada revisión del estado del arte en torno a métodos, técnicas y herramientas para la gestión de procesos de negocio y las arquitecturas orientadas a componentes y servicios

Valida los aportes mediante un caso de estudio real y se obtiene como resultado una visión mejorada de TesinaFlow, el proceso que realiza el seguimiento de las

tesinas de grado que los alumnos deben realizar para obtener un grado académico en la Facultad de Informática.

Estas contribuciones se reflejaron en las siguientes publicaciones que se detallan divididas en dos grupos: 1) publicaciones referidas a la mejora de MISP en torno al modelado de procesos y servicios y su integración y 2) publicaciones vinculadas a los avances tecnológicos a aplicar en el despliegue, ejecución y monitoreo de procesos de negocio.

### **Publicaciones vinculadas a la mejora de MISP**

- *Análisis de Tecnologías para Implementar un Marco Integrador de SOA y BPM* [99] – El trabajo realiza un análisis de las tecnologías existentes que puedan ser capaces de dar soporte a un marco metodológico integrador de procesos y servicios. Publicado en ASSE 2010 - 11th Argentine Symposium on Software Engineering”. 39° Jornadas Argentinas de Informática. 2010.
- *Process - Service Interaction Using an SOA-BPM Methodology* [91] – El trabajo se focaliza en mostrar la interacción entre las actividades de un proceso de negocio y los servicios. Se presenta un metamodelo que a soporte a esta interacción e implementa un prototipo de editor para dicho metamodelo. Publicado en XXX Conferencia Internacional de la Sociedad Chilena de Ciencia de la Computación (SCCC'2011). Curicó, Chile. ISBN 978-0-7695-4689-6. 2011.
- *Integrating Process and Services Through Meta-Models* [124] – El trabajo se basa en el uso de metamodelos como medio para definir sintaxis sin ambigüedades para modelar servicios e integrarlo con el metamodelo SCA para describir componentes. Publicado en International Conference WWW/Internet 2012 organizado por International Association for Development of the Information Society (IADIS). Madrid, España, ISBN 978-989-8533-11-1. 2012.
- *Conceptualización de Servicios dentro de una Metodología SOA/BPM* [12] – Este artículo presenta una propuesta para la integración de un metamodelo de servicios – propuesto en un trabajo anterior e integrado con el metamodelo BPMN; y un meta-modelo de componentes – el definido por el estándar SCA (Service Component Architecture). La contribución de este trabajo permite formalizar la interacción entre dos etapas de la metodología SOA/BPM. Publicado en CLEI 2012 XXXVIII Conferencia Latinoamericana en Informática. Medellín, Colombia. ISBN 9781467307949. 2012.
- *Formalizing the Conceptualization of Services and their Relationships with Software Components* [[13] – Este trabajo propone la integración entre dos estándares SCA, para describir componentes y WSDL para especificar nuevos servicios. Esta integración contribuye a completar la mejora de MISP. Publicado en XXXII Conferencia Internacional de la Sociedad Chilena de Ciencias de la Computación (SCCC 2013). Temuco, Chile. ISBN 978-956-7019-95-3. 2013.

- *Sistema basado en BPM para el Seguimiento del Proceso Licitatorio y la Ejecución de Proyectos del Programa PMGM-UEC-Ministerio del Interior y Transporte de la Nación Argentina* [149] – El trabajo describe la experiencia de implementación de un sistema basado en BPM para el seguimiento del proceso licitatorio en una organización pública y muestra las mejoras obtenidas como consecuencia de dicha implementación. La implementación se llevó a cabo utilizando las directivas propuestas por MISP. Publicado en Simposio Informática en el Estado, en el marco de las 43° Jornadas Argentinas de Informática, Simposio de Informática en el Estado. ISSN 1851-2526. 2014.

#### **Publicaciones vinculadas a la aplicación de avances tecnológicos.**

- *Conceptos de Dinamismo Aplicados a Servicios y Workflows en BPMS Basados en Cloud Computing* [146] – El trabajo presenta una serie de conceptos en la bibliografía actual que comienzan a reconsiderarse al insertar el paradigma de BPM dentro de la nube, así como propuestas de implementación para varios de ellos en un BPMS (Business Process Management System) particular presente en la industria. Publicado en XV Workshop de Investigadores en Ciencia de la Computación: WICC 2013. Paraná, Entre Ríos, Argentina. ISBN 978-987-28179-6-1. 2013.
- *Constructing and Monitoring Processes in BPM Using Hybrid Architectures* [130] [144] (presentado en la conferencia internacional e invitado a ser incluido en la revista de la editorial asociada) – Este trabajo analiza conceptos de la bibliografía actual y propone una arquitectura para un sistema de monitoreo de procesos distribuidos, considerando aspectos de transparencia de la localización y rastreo de instancias de procesos en la nube. Publicado en IJACSA (International Journal of Advanced Computer Science and Applications) U.S ISSN: 2156-5570(Online). U.S ISSN: 2158-107X (Print). 2013 y presentado en SAI Science and Information Conference 2013, London, UK 2013.
- *Monitoreo de Procesos Distribuidos en el Cloud. Una Propuesta Arquitectónica* [145] – El trabajo propone una arquitectura para un sistema de monitoreo de procesos distribuidos, considerando distintos factores de diseño, tales como la transparencia de localización y los datos necesarios para el rastreo de instancias en el sistema en la nube. Publicado en WBPM 2013: Chilean Workshop on Business Process Management (BPM). Temuco, Chile. ISBN 978-956-7019-95-3. 2013.
- *Ejecución y Monitoreo de Procesos de Negocios Distribuidos entre Diferentes Motores de Bonita OS* [140] – El trabajo propone una implementación de una arquitectura para un sistema de monitoreo de procesos distribuidos utilizando Bonita Open Solution como motor de procesos, su API y el uso de conectores personalizados. Publicado en XVI Workshop de Investigadores en Ciencia de la Computación: WICC 2014. Ushuaia, Tierra del Fuego, Argentina. ISBN 978-950-34-1084-4. 2014.

- *Socializar el Despliegue y Seguimiento de Procesos de Negocio* [147] - En este trabajo se analiza el impacto de las interacciones entre usuarios para seguimiento y monitoreo de procesos de negocio menos estructurados y se lo aplica extendiendo un motor de procesos clásico con características sociales. Este prototipo, denominado SocialFlow, incorpora características sociales las utiliza para enriquecer el seguimiento de los procesos de negocio. Publicado en WBPM 2014: Chilean Workshop on Business Process Management (BPM). Noviembre, 2014. Talca, Chile. 2014.

Las propuestas de trabajo para llevar adelante esta tesis, asistieron a la concreción de las siguientes tesinas de grado y trabajos de cátedra de las carreras de Licenciatura en Informática y Licenciatura en Sistemas, todas bajo la dirección y coordinación de la autora de esta tesis:

- *Ejecución y Monitoreo de Procesos de Negocio Distribuidos entre Diferentes Motores de Bonita Open Solution* (2014 en prensa), tesina desarrollada por el alumno Leonardo Karabagosian para obtener el grado de Licenciado en Sistemas de la Facultad de Informática de la UNLP. El objetivo de esta tesis es investigar y elaborar un marco de desarrollo y ejecución para la ejecución de procesos de negocios distribuidos en diferentes motores de proceso de Bonita Open Solution para que se comporten como si se tratase de un solo proceso, y a la vez, brindar una herramienta de monitoreo distribuido basado en servicios web para el proceso de negocio unificado.
- *Aplicación de Técnicas de Process Mining para Análisis de Procesos de Negocio Desplegados en un BPMS* (2014) tesina desarrollada por la alumna Virginia Magliano para obtener el grado de Licenciado en Sistemas de la Facultad de Informática de la UNLP. El objetivo de la tesina radica principalmente en el análisis y la propuesta de un enfoque sobre Process Mining como medio para analizar los procesos de negocio ya implantados en un BPMS. La implementación de este enfoque mediante el uso de conectores de Bonita Open Solution constituye el principal aporte de la tesina para esta tesis doctoral.

### 7.3. TRABAJOS FUTUROS

A partir de los resultados obtenidos de esta tesis es posible definir las siguientes áreas de trabajo futuro: 1) la definición de lenguajes para modelar procesos y servicios que consideren tanto aspectos de distribución como de socialización de procesos de negocio, 2) el estudio de mejoras en las técnicas de monitoreo de procesos de negocio tanto para absorber nuevos rastros como para afrontar la ejecución distribuida y 3) la integración de los motores de procesos dentro de portales, mediante conectores genéricos que permitan integrar a la gestión por procesos con el resto de las actividades que desarrolla una organización.

En la definición de lenguajes para modelar procesos y servicios que consideren la distribución como la socialización de procesos de negocio, existen en la

actualidad algunas propuestas de extensión de BPMN y analizadas en el Capítulo 3, donde se plantea la inclusión de nuevos símbolos para representar características sociales, pero estas propuestas derivan en una aplicación Web tradicional. Los trabajos futuros aquí definidos plantean hacer uso de esas extensiones para trasladarlas, mediante integración de modelos, a los motores de procesos y vincularlas a las etapas de despliegue, ejecución y monitoreo. En el mismo sentido, se consideran realizar extensiones rotacionales para incluir la distribución de procesos desde las etapas de modelado de procesos y servicios.

En cuanto a las mejoras en las técnicas de monitoreo de procesos de negocio, se propone analizar distintos mecanismos para enriquecer los rastros de ejecución de los procesos mediante OLC, demonios o agentes que estampen información más detallada y que además sean aptos en entornos distribuidos.

Por último, la integración de los motores de procesos dentro de portales, constituye un área interesante a la hora de instrumentar la gestión por procesos dentro de una organización dado que plantea un esquema centralizado de gestión donde no se evidencia la existencia de un *workflow* sino que el trabajo a nivel operativo se desarrolla en un único entorno integrado a las aplicaciones de uso frecuente no necesariamente vinculada a las actividades de los procesos de negocio.

En lo relativo a las mejoras en las técnicas de monitoreo de procesos de negocio, se cuenta a la fecha con algunas publicaciones vinculadas:

- *Técnicas de Ejecución y Monitoreo de procesos en Cloud BPM* [151] - En el presente trabajo se analizan conceptos asociados a la descomposición de procesos, así como al monitoreo de las actividades y el uso de los objetos de datos generados a través de sus transiciones. A su vez se enuncian las arquitecturas y metodologías implementadas para el desarrollo de estos conceptos. Publicado en XVI Workshop de Investigadores en Ciencia de la Computación 2014. WICC 2014. Ushuaia, Tierra del Fuego, Argentina. ISBN 978-950-34-1084-4. 2014.
- *OLC y Monitoreo de Procesos en el Cloud: un Caso de Estudio* [148] - En el presente trabajo se aplica una arquitectura de ejecución y monitoreo de procesos descompuestos definida en trabajos previos sobre un caso de estudio concreto en un entorno de investigación, empleando a su vez objetos de datos OLC (Objetos de ciclo de vida) para la ampliación de los resultados obtenidos. Publicado en WBPM 2014: Chilean Workshop on Business Process Management (BPM). Noviembre, 2014. Talca, Chile. 2014.
- *Decomposed Processes in Cloud BPM: Techniques for Monitoring and The Use of OLC* [150] - En este trabajo se presenta un modelo de monitoreo de procesos distribuidos que considera OLC (Object Life Cycle) para proveer un amplio conjunto de información con el propósito de medir y mejorar procesos de negocio. También compara características de ejecución y monitoreo tanto en ambientes híbridos como embebidos, incluyendo el uso de OLC. Publicado en SERP'14:2014 International Conference on Software

Engineering Research and Practice. Las Vegas, Nevada, USA. ISBN 1-60132-286-0. 2014



## REFERENCIAS

- [1] Bazán P. "Un modelo de integrabilidad con SOA y BPM". Tesis de Maestría en Redes de Datos. Facultad de Informática. Universidad Nacional de La Plata. Abril 2010
- [2] Juric Matjaz B., Loganathan Ramesh, Poornachandra Sarang, Frank Jennings "SOA Approach to Integration XML, Web services, ESB, and BPEL in real-world SOA projects". Packt Publishing. ISBN 978-1-904811-17-6. 2007
- [3] Bruno, G., Dengler, F., Jennings, B., Khalaf, R., Nurcan, S., Prilla, M., & Silva, R. "Key challenges for enabling agile BPM with social software". *Journal of Software Maintenance and Evolution: Research and Practice*, 23(4), 297-326. 2011.
- [4] Erol, S., Granitzer, M., Happ, S., Jantunen, S., Jennings, B., Johannesson, P., & Schmidt, R. "Combining BPM and social software: contradiction or chance?". *Journal of software maintenance and evolution: research and practice*, 22(6- 7), 449-476. 2010.
- [5] Oracle White Paper. "Business Process Management, Service-Oriented Architecture, and Web 2.0: Business Transformation or Train Wreck?". 2008. <http://whitepapers.techrepublic.com.com/abstract.aspx?docid=992727> (al 23/11/2009)
- [6] Kahn Rashid. "Social Networking and BPM of the Future". *BpTrends Column*. [http://bptrends.com/publicationfiles/TEN-COL-BPM-A\\_Global\\_View-Social%20Networking%20and%20BPM%20of%20the%20Future.pdf](http://bptrends.com/publicationfiles/TEN-COL-BPM-A_Global_View-Social%20Networking%20and%20BPM%20of%20the%20Future.pdf) (al 23/11/2009)
- [7] Abbattista, F., Calefato, F., Gendarmi, D., Lanubile, F.: "Incorporating Social Software into Agile Distributed Development Environments". *Proceedings of the 23rd Automated Software Engineering (ASE) workshop*, pp.46-51. 2008
- [8] Rangihia, M. E., & Karakostas, B. "Goal-driven social business process management". *Science and Information Conference (SAI)*, 2013 (pp. 894-901). IEEE. 2009.
- [9] <http://www.w3.org/TR/ws-cdl-10/>
- [10] <http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.pdf>
- [11] Bazan Patricia, Gabriela Perez, Roxana Giandini, and Javier Diaz. "Process-Service Interactions Using a SOA-BPM-Based Methodology". *Proceedings of the 2011 30th International Conference of the Chilean Computer Science Society (SCCC '11)*. IEEE Computer Society, Washington, DC, USA, 100-107. DOI=10.1109/SCCC.2011.14 <http://dx.doi.org/10.1109/SCCC.2011.14>
- [12] Bazan P., Giandini R., Perez G., Estevez E., Diaz J., "Conceptualización de Servicios dentro de una Metodología SOA/BPM". *CLEI 2012 XXXVIII Conferencia Latinoamericana en Informática*. Medellín, Colombia. ISBN 9781467307949. 2012.
- [13] Bazan P., Giandini R., Perez G., Estevez E., Diaz J. "Formalizing the conceptualization of services and their relationships with software components". *XXXII Conferencia Internacional de la Sociedad Chilena de Ciencias de la Computación (SCCC 2013)*. Temuco, Chile.. ISBN 978-956-7019-95-3. 2013
- [14] Weske Mathias, "Business Process Management: Concepts, Languages, Architectures". Springer, Pag 4-21 y 73-80. ISBN 978-3-540-73521-2. 2008
- [15] Kiran Garimella, Michael Lees, Bruce Williams."BPM For Dummies". Wiley Publishing, Inc. Pag. 5-61 ISBN- 978-0-470-28571-8. 2008
- [16] Xoan Pardo "Curso Cloud Computing" dictado en el marco de la carrera del Doctorado en Ciencias Informáticas de la Facultad de Informática de la UNLP. 2013
- [17] <http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>
- [18] Informe ENISA (European Network and Information Security Agency): "Cloud Computing: beneficios, riesgos y recomendaciones para la seguridad de la información". <http://www.enisa.europa.eu/activities/risk-management/files/deliverables/cloud-computing-risk-assessment>
- [19] Antonio Fumero y Genís Roca. "Web 2.0". Con la colaboración de Fernando Saez Vaca. Este libro se publica bajo licencia Creative Commons de tipo "Reconocimiento-NoComercial-SinObraDerivada". [http://fundacionorange.es/areas/25\\_publicaciones/WEB\\_DEF\\_COMPLETO.pdf](http://fundacionorange.es/areas/25_publicaciones/WEB_DEF_COMPLETO.pdf) (a marzo 2014)

- [20] Dollmann, T., Fettke, P., Loos, P., & Vanderhaeghen, D. Web 2.0 enhanced automation of collaborative business process model management in cooperation environments. 2009.
- [21] <http://www.islandone.org/Foresight/WebEnhance/HPEK1.html>
- [22] BOYD, S., "Are you ready for social software?", *Darwin Magazine*, Spring, 2003.
- [23] User Generated Content, Social Media, and Advertising — An Overview. 2008. [http://www.iab.net/media/file/2008\\_ugc\\_platform.pdf](http://www.iab.net/media/file/2008_ugc_platform.pdf) (a junio 2014)
- [24] T. Anstett, F. Leymann, R. Mietzner, and S. Strauch, "Towards bpel in the cloud: Exploiting different delivery models for the execution of business processes," Proceedings of the 2009 Congress on Services - I. Washington, DC, USA: IEEE Computer Society, pp. 670–677. 2009
- [25] T. Kirkham, S. Winfield, T. Haberecht, J. Müller, G. De Angelis, "The Challenge of Dynamic Services in Business Process Management", University of Nottingham, United Kingdom, Springer, 2011
- [26] M. Minor, R. Bergmann, S. Görg, "Adaptive Workflow Management in the Cloud – Towards a Novel Platform as a Service", Business Information Systems II, University of Trier, Germany, 2012
- [27] M Mevius, R. Stephan, P. Wiedmann, "Innovative Approach for Agile BPM", eKNOW 2013: The Fifth International Conference on Information, Process, and Knowledge Management, 2013.
- [28] Evert Duipmans, Dr. Luis Ferreira Pires, "Business Process Management in the cloud: Business Process as a Service (BPaaS)", University of Twente, April, 2012.
- [29] M. Gerhards, V. Sander, A. Belloum, "About the flexible Migration of Workflow Tasks to Clouds - Combining on and off premise Executions of Applications", CLOUD COMPUTING 2012: The Third International Conference on Cloud Computing, GRIDs, and Virtualization, 2012.
- [30] Hubert Scheuerlein, Falk Rauchfuss, Yves Dittmar, Rüdiger Molle, Torsten Lehmann, Nicole Pienkos, Utz Settmacher, "New methods for clinical pathways –Business Process Modeling Notation (BPMN) and Tangible Business Process Modeling (t.BPM)". Springer-Verlag 2012.
- [31] Marco Brambilla, Piero Fraternali, Carmen Vaca, Stefano Butti, "Combining Social Web and BPM for Improving Enterprise Performances: the BPM4People Approach to Social BPM", WWW 2012, European Projects Track, Abril 16–20, Lyon, France, 2012.
- [32] S. Balzert, P. Fettke, P. Loos, "Enhancement of traditional Business Process Management with reflection – a new perspective for Organizational Learning", Institute for Information Systems (IWi) at German Research Center for Artificial Intelligence (DFKI), Germany, 2012.
- [33] Jiri Kolar, Tomas Pitner, "Agile BPM in the age of Cloud technologies", Scalable Computing: Practice and Experience, 2012.
- [34] Andreas Lehmann and Dirk Fahland, "Information Flow Security for Business Process Models - just one click away", University of Rostock, Germany, 2012.
- [35] Aleš Frece, Gregor Srdić, Matjaž B. Jurič, "BPM and iBPMS in the Cloud", Proceedings of the 1st International Conference on Cloud Assisted ServiceS, Bled. 2012
- [36] Dr. Manuel Goetz, "Integration of Business Process Management and Complex Event Processing", Germany, 2012.
- [37] Marielba Zacarias, Paula Ventura Martins, "Collaborative methods for Business Process Discovery", Portugal, Springer-Verlag 2012.
- [38] Rafael Accorsi, Thomas Stocker, Günter Müller, "On the Exploitation of Process Mining for Security Audits: The Process Discovery Case", Department of Telematics, University of Freiburg, Germany, 2012.
- [39] Pons, C., Giandini, R., & Pérez, G. "Desarrollo de Software Dirigido por Modelos. Conceptos teóricos y su aplicación práctica". *EDULP & McGraw-Hill Educación*. 2010.
- [40] Bruning, J.; Gogolla, M. "UML Metamodel-based Workflow Modeling and Execution," Enterprise Distributed Object Computing Conference (EDOC), IEEE, vol., no., pp.97-106, URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6037564&isnumber=60375532006>. 2011
- [41] Kalnins, A., Vitolins, V., "Use of UML and model transformations for workflow process definitions" Databases and Information Systems, Baltic DB&IS'2006, edited by Olegas Vasilecas, Johann Eder, Albertas Caplinskas, Vilnius, Technika, pp. 3-15. 2006.
- [42] Brüning, J., Gogolla, M., Forbrig, P., "Modeling and formally checking workflow properties using UML and OCL". Perspectives in Business Informatics Research, Lecture Notes in Business Information Processing, Springer Berlin Heidelberg, ISBN 978-3-642- 16101-8, vol 64, pp. 130 a 145, URL: [http://dx.doi.org/10.1007/978-3-642-16101-8\\_112009](http://dx.doi.org/10.1007/978-3-642-16101-8_112009). 2010
- [43] Gogolla, M., Büttner, F., Richters, M., "USE: A UML-Based Specification Environment for Validating", Science of Computer Programming, vol 69, (2007), pp 27–34.

- [44] SCA Assembly Model Specification 1.1, Open SOA Collaboration. 2009.
- [45] Business Process Model and Notation (BPMN) 2.0, Beta 1, OMG, May 2009.
- [46] Zhaogang, H., Li, Z., "From UML 2.0 Activity Diagram to YAWL: The Controlflow Aspect", ICEES 2011, 14–16 Singapore. 2011
- [47] Kühne, S., Kern, H., Gruhn, V., Laue, R., "Business process modeling with continuous validation", *Journal of Software Maintenance and Evolution: Research and Practice*, 2010, vol 22, issue 6-7, pp 547- 566, DOI: 10.1002/smr.517. 2010
- [48] Dahman, K., Charoy, F., Godart, C., "Generation of Component Based Architecture from Business Processes: Model Driven Engineering for SOA", *IEEE 8th European Conference on Web Services (ECOWS)*, pp.155-162, URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5693257&isnumber=5693237>. 2010
- [49] Buchwald, S., Bauer, T., Reichert, M., "Bridging the Gap Between Business Process Models and Service Composition Specifications". In: *Service Life Cycle Tools and Technologies: Methods, Trends and Advances*, pp. 124-153. ISBN 978-1613501597. 2011.
- [50] Arsanjani, A., Ghosh, S., Allam, A., Abdollah, T., Ganapathy, S., Holley, K., "SOMA: a method for developing service-oriented solutions", *IBM System Journal*, 47 3. 2008.
- [51] OMG, Service Oriented Architecture Modeling Language (SoaML) – Specification for the UML Profile and Metamodel for Services (UPMS). OMG document: ad/2008-08-04, URL: <http://www.omg.org/docs/ad/08-08-04.pdf>. 2009.
- [52] De Castro, M., Wieringa, "Towards a service-oriented MDA-Based approach to the alignment of business process with it systems: from the business model to a Web Services composition model", *International Journal of Cooperative Information Systems (IJCIS)*, vol 18, issue 2, pp. 225-260, DOI: 10.1142/S0218843009002038. 2009.
- [53] Hahn, C., Dmytro, P., Fischer, K., "A Model-Driven Approach to Close the Gap between Business Requirements and Agent-Based Execution". *Proceedings of the 4th Workshop on Agent-based Technologies and applications for enterprise interoperability*, Toronto, Canada, 2010.
- [54] Sabraoui, A., Ennouaary, A., Khriess, I., El Koutbi, M. "An MDA-Based Approach for WS Composition Using UML Scenarios" 2012. 2012 Ninth International Conference on Information Technology- New Generations. Pag 306-313. ISBN 978-1-4673-0798-7. 2012.
- [55] El Hog, C., Djemaa, R.B., Amous, I. AWS-WSDL: A WSDL extension to support adaptive web service. *ACM International Conference Proceeding Series*, pp. 477-480. 2011.
- [56] Mardiana, M., Araki, K., Omori, Y. "MDA and SOA Approach to Development of Web Application Interface" (2011). *TENCON 2011 IEEE*, Pag 226-231 ISBN 978-1-4577-0256-3. 2011.
- [57] Bruning, J., & Gogolla, M. "UML Metamodel-based Workflow Modeling and Execution". In *Enterprise Distributed Object Computing Conference (EDOC)*, 2011 15th IEEE International (pp. 97-106). IEEE. 2011.
- [58] Han, Z., & Zhang, L. From UML 2.0 Activity Diagram to YAWL: The Controlflow Aspect. *Energy Procedia*, 11(Complete), 1751-1758. 2011.
- [59] Gogolla, M., Büttner, F., & Richters, M. USE: A UML-based specification environment for validating UML and OCL. *Science of Computer Programming*, 69(1), 27-34. 2007.
- [60] Kalnins, A., & Vitolins, V. Use of UML and model transformations for workflow process definitions. *arXiv preprint cs/0607044*. 2006
- [61] Gallaba, K., Pavalanathan, U., Jayawardena, I., Sooriyabandara, E., & Nanayakkara, V. Levi–A Workflow Engine Using BPMN 2.0. In *Web Information Systems Engineering–WISE 2011 and 2012 Workshops* (pp. 118-130). Springer Berlin Heidelberg. 2013.
- [62] Kühne, S., Kern, H., Gruhn, V., & Laue, R. Business process modeling with continuous validation. *Journal of Software Maintenance and Evolution: Research and Practice*, 22(6- 7), 547-566. 2010.
- [63] Mili, H., Tremblay, G., Jaoude, G. B., Lefebvre, É. Elabed, L., & Boussaidi, G. E. Business process modeling languages: Sorting through the alphabet soup. *ACM Computing Surveys (CSUR)*, 43(1), 4. 2010.
- [64] Ali Arsanjani, Ph.D. "Service-oriented modeling and architecture". <http://www.ibm.com/developerworks/webservices/library/ws-soa-design1/>. 2004. (al 16/10/2009)
- [65] Olaf Zimmermann, Pal Krogdahl, Clive Gee. "Elements of Service-Oriented Analysis and Design". <http://www-128.ibm.com/developerworks/library/ws-soa-design1/>. 2004. (al 16/10/2009)
- [66] Olaf Zimmermann, Frank Mueller. "Web Services project roles". <http://www-128.ibm.com/developerworks/library/ws-roles/>. 2004. (al 16/10/2009)
- [67] Pulier Eric, Taylor Hugo. "Understanding Enterprise SOA". Manning Publications Co. ISBN 1-932394-59-1. Pag 1-73. 2006.

- [68] Selmeçi, A., Orosz, T. "Usage of SOA and BPM changes the roles and the way of thinking in development" IEEE 10th Jubilee International Symposium on Intelligent Systems and Informatics, SISY 2012, art. no. 6339526, pp. 265-271. <http://www.scopus.com/inward/record.url?eid=2-s2.0-84870675074&partnerID=40&md5=0f8b2e6f95df1845b26b538516e70b8d>. 2012.
- [69] Chen, P. Y., Hwang, S. Y., & Lee, C. H. A Dynamic Service Composition Architecture in Supporting Reliable Web Service Selection. In Service Science and Innovation (ICSSI), 2013 Fifth International Conference on (pp. 237-238). IEEE.2013.
- [70] Belli, F., Endo, A. T., Linschulte, M., & Simao, A. A holistic approach to model- based testing of Web service compositions. *Software: Practice and Experience*, 44(2), 201-234. 2014.
- [71] Wu, C. S., & Huang, C. H. The Web Services Composition Testing Based on Extended Finite State Machine and UML Model. In Service Science and Innovation (ICSSI), 2013 Fifth International Conference on (pp. 215-222). IEEE. 2013.
- [72] Gu, Q., & Lago, P. Service identification methods: a systematic literature review. In *Towards a Service-Based Internet* (pp. 37-50). Springer Berlin Heidelberg. 2010.
- [73] Zadeh, Ali Taei, et al. "A Systematic Input Selection for Service Identification in SMEs." *Journal of Applied Sciences* 12.12: 1232-1244. 2012.
- [74] Grzech, Adam, et al. "Specifications and Deployment of SOA Business Applications Within a Configurable Framework Provided as a Service." *Advanced SOA Tools and Applications*. Springer Berlin Heidelberg. 7-71. 2014.
- [75] Fang, Zhenyu, and Changqing Yin. "BPM architecture design based on cloud computing." 2010.
- [76] Frece, Aleš, Gregor Srdić, and Matjaž B. Jurič. "BPM and iPMS in the Cloud." *The 1st International Conference on Cloud Assisted Services*. 2012.
- [77] Duipmans, Evert F., Luis Ferreira Pires, and Luiz Olavo Bonino da Silva Santos. "Towards a BPM cloud architecture with data and activity distribution." *Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2012 IEEE 16th International*. IEEE, 2012.
- [78] Cabanillas, Cristina, et al. "Towards the Enhancement of Business Process Monitoring for Complex Logistics Chains." *Business Process Management Workshops*. Springer International Publishing, 2014.
- [79] Herzberg, Nico, and Andreas Meyer. "Improving Process Monitoring and Progress Prediction with Data State Transition Events." *ZEUS*. 2013.
- [80] Y.-B. Han, J.-Y. Sun, G.-L. Wang, and H.-F. Li, "A cloud-based bpm architecture with user-end distribution of non-compute-intensive activities and sensitive data," *J. Comput. Sci. Technol.*, vol. 25, no. 6, pp.1157–1167, 2010.
- [81] W. Fdhila, U. Yildiz, and C. Godart, "A flexible approach for automatic process decentralization using dependency tables," in *ICWS*, pp. 847–855. 2009.
- [82] [M. Minor, R. Bergmann, S. Görg, "Adaptive Workflow Management in the Cloud – Towards a Novel Platform as a Service", *Business Information Systems II*, University of Trier, Germany, 2012
- [83] T. Dornemann, E. Juhnke, and B. Freisleben, "On demand resource provisioning for bpm workflows using amazon's elastic compute cloud," in *Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid*, ser. CCGRID '09. Washington, DC, USA: IEEE Computer Society, pp. 140–147. 2009.
- [84] Huang, H., Zhang, Y. L., Zhang, M., & Computing, C. A Survey of Cloud Workflow. Published by Atlantis Press, Paris, France. © the authors, 2013
- [85] T. Anstett, F. Leymann, R. Mietzner, and S. Strauch, "Towards bpm in the cloud: Exploiting different delivery models for the execution of business processes," in *Proceedings of the 2009 Congress on Services - I*. Washington, DC, USA: IEEE Computer Society, pp. 670–677. 2009.
- [86] Liu, Hua, et al. "Towards simplifying and automating business process lifecycle management in hybrid clouds." *Web Services (ICWS), 2012 IEEE 19th International Conference on*. IEEE, 2012.
- [87] Massonet, Philippe, et al. "A monitoring and audit logging architecture for data location compliance in federated cloud infrastructures." *Parallel and Distributed Processing Workshops and Phd Forum (IPDPSW), 2011 IEEE International Symposium on*. IEEE, 2011.
- [88] Roder, A., Matthias Lehmann, and Klaus Kabitzsch. "Monitoring service choreographies." *Industrial Informatics (INDIN), 2011 9th IEEE International Conference on*. IEEE, 2011.
- [89] Díaz, Javier F., Bazan P. Queiruga C, Rodriguez F, Gotelli G. "Entornos para usar BPM en aplicaciones JAVA: un análisis comparativo." *XI Workshop de Investigadores en Ciencias de la Computación*. 2009.
- [90] Fingar P. "Strategic business process management in the cloud. Optimize business processes in the cloud with intelligent business platform" IBM Corporation Software Group. 2014.

- [91] Bazan P., Perez G, Giandini R., Diaz J. "Process - service interaction using an SOA-BPM methodology" XXX Conferencia Internacional de la Sociedad Chilena de Ciencia de la Computación (SCCC'2011). Curico, Chile. ISBN 978-0-7695-4689-6. 2011.
- [92] Imaz M.: Curso Introducción a SOA/BPM. Organizado por SADIO (2008).
- [93] IBM Global Technology Services: How service-oriented architecture (SOA) impacts your IT infrastructure 2008.
- [94] Gustavo Giorgetti. "Transformando. Capitulo: Administrando la Complejidad." Editorial Eudeba ISBN 950-231289-9. 2003.
- [95] Object Management Group (OMG), <http://www.omg.org>.
- [97] Business Process Modeling Notation (BPMN) Version 1.2 OMG, <http://www.omg.org/spec/BPMN/1.2>.
- [98] Unified Modeling Language (UML), version 2.2 OMG, <http://www.omg.org/technology/documents/formal/uml.htm>.
- [99] Bazán P., Giandini R. Diaz J. "Análisis de tecnologías para implementar un marco integrador de SOA y BPM" ASSE 2010 - 11th Argentine Symposium on Software Engineering". 39° Jornadas Argentinas de Informática. 2010.
- [100] Erl, Thomas. "SOA Principles of Service Design". Prentice Hall. ISBN-13: 9780132344821. 2007. Pag.25-119.
- [101] [Bell Michel. "Service-Oriented Modeling. Services Analysis, Design and Architecture". John Wiley & Sons, Inc., ISBN 978-0-470-14111-3. 2008.
- [102] <http://www.e3value.com/>.
- [103] Joris Hulstijn and Jaap Gordijn. Risk analysis for inter-organizational controls. In Joaquim Filipe and Jose Cordeiro editors, Proceedings of the 12th International Conference on Enterprise Information Systems (ICEIS 2010), Vol. 3:314-320, 2010.
- [104] SOA Tools Platform Project <http://www.eclipse.org/stp/>.
- [105] [Rational Software Modeler <http://publib.boulder.ibm.com/infocenter/rsmhelp/v7r5m0/index.jsp?topic=/com.ibm.xtools.transform.cfm.wbm.doc/topics/ccwmbmtransf.html>. (al 16/10/2009).
- [106] Transformation to SOA: Part 1. From business process to service model architecture using IBM WebSphere Business Modeler and IBM Rational Software Architect [http://www.ibm.com/developerworks/rational/library/07/1225\\_bennett-balaba/](http://www.ibm.com/developerworks/rational/library/07/1225_bennett-balaba/). 2007 (al 16/10/2009).
- [107] Marco Brambilla, Stefano Butti, Piero Fraternali: WebRatio BPM: A Tool for Designing and Deploying Business Processes on the Web. ICWE 2010: 415-429. 2010.
- [108] Marco Brambilla, Stefano Ceri, Piero Fraternali, Ioana Manolescu: Process modeling in Web applications. ACM Trans. Softw. Eng. Methodol. 15(4): 360-409. 2006.
- [109] Delgado, A., Ruiz, F., García - Rodríguez de Guzmán, I., Piattini, M.: Towards a Service- Oriented and Model-Driven framework with business processes as first-class citizens, In: 2nd Int. Conf. on Business Process and Services Computing (BPSC'09), Leipzig, 2009.
- [110] Delgado, A., García - Rodríguez de Guzmán, I., Ruiz, F., Piattini, M.: Tool support for Service Oriented development from Business Processes, 2nd International Workshop on Model-Driven Service Engineering (MOSE'10), Málaga, 2010.
- [111] SOA Tools BMPN Modeler <http://www.eclipse.org/bpmn/>
- [112] Erradi, A., Anand, S., Kulkarni, N. N.: SOAF: An Architectural Framework for Service Definition and Realization, IEEE Int. Conf. on Services Computing, pp. 151-158. 2006.
- [113] Roxana Giandini, Gabriela Pérez, Claudia Pons. Un lenguaje de Transformación específico para Modelos de Proceso del Negocio. XXXVI Conferencia Latinoamericana de Informática (CLEI 2010). Asunción, Paraguay. 2010.
- [114] SCA Assembly Model Specification 1.1, Open SOA Collaboration, 2009.
- [115] Advancing open standars for the information society, URL:<http://www.oasis-open.org/>.
- [116] Herramientas para SCA (Service Component Architecture) para la plataforma Eclipse, URL: <http://www.eclipse.org/soa/sca/>.
- [117] Eclipse Modeling Framework EMF. <http://www.eclipse.org/modeling/emf/>
- [118] The Eclipse Project. Home Page. Copyright IBM Corp, 2000.
- [119] Eclipse EuGENia. [www.eclipse.org/gmt/epsilon/doc/eugenia/](http://www.eclipse.org/gmt/epsilon/doc/eugenia/)
- [120] <http://www.w3.org/TR/wsd/>
- [121] <http://www.w3.org/TR/2007/REC-wsdl20-20070626>

- [122] El Hog, C., Djemaa, R.B., Amous, I. AWS-WSDL: A WSDL extension to support adaptive web service. ACM International Conference Proceeding Series, pp. 477-480. 2011.
- [123] V. de Castro, Marcos, E., Vela, B., "Representing WSDL with extended UML". Revista Colombiana de Computation, vol. 5, 2004.
- [124] Bazan P., Giandini R., Perez G., Estevez E., Diaz J. "Integrating process and services through meta-models" IADIS International Conference IADIS WWW/Internet 2012 organizado por International Association for Development of the Information Society (IADIS). Madrid, España, ISBN 978-989-8533-11-1. 2012.
- [125] Stoitsev, Vassil, and Paul Grefen. Business process technology and the cloud: Defining a business process cloud platform. Beta Working Paper Series 393, School of Industrial Engineering, Eindhoven University of Technology, 2012.
- [126] IBM. IBM Cloud Computing Reference Architecture 2.0. Document. 2011.
- [127] OASIS. OASIS UDDI Specification. [Online]. <https://www.oasis-open.org/committees/uddi-spec/faq.php>
- [128] S Balzert, P Fettke, P Loos, "A Framework for Reflective Business Process Management", 45th Hawaii International Conference on System Sciences, USA, 2012.
- [129] Dr. Luis Ferreira Pires, "Business Process Management in the cloud: Business Process as a Service (BPaaS)", University of Twente, April, 2012.
- [130] Martinez Garro J., Bazan P. "Constructing and monitoring processes in BPM using hybrid architectures" IJACSA (International Journal of Advanced Computer Science and Applications) U.S ISSN: 2156-5570(Online). U.S ISSN: 2158-107X (Print). 2013. <http://thesai.org/Publications/ViewPaper?Volume=3&Issue=4&Code=SpecialIssue&SerialNo=10>
- [131] Marco Brambilla, Piero Fraternali, and Carmen Vaca, "BPMN and Design Patterns for Engineering Social BPM Solutions", Politecnico di Milano, Piazza L. da Vinci 32, Milano, Italy, 2012
- [132] Marco Brambilla, Piero Fraternali, Carmen Vaca, Stefano Butti, "Combining Social Web and BPM for Improving Enterprise Performances: the BPM4People Approach to Social BPM", WWW 2012, European Projects Track, Abril 16–20, Lyon, France, 2012.
- [133] Bonita Open Solution <http://es.bonitasoft.com/>. October, 2013.
- [134] S. Balzert, P. Fettke, P. Loos, "Enhancement of traditional Business Process Management with reflection – a new perspective for Organizational Learning", Institute for Information Systems (IWi) at German Research Center for Artificial Intelligence (DFKI), Germany, 2012.
- [135] Creación de un conector en Bonitasoft - <http://documentation.bonitasoft.com/creating-connector>.
- [136] Java Authentication and Authorization Service (JAAS) (Agosto 2013) - <http://docs.oracle.com/javase/6/docs/technotes/guides/security/jaas/JAASRefGuide.html>
- [137] Toàn Nguyễn, Jean-Antoine-Désidéri, "Resilience Issues for Application Workflows on Clouds", Project OPALE, INRIA Grenoble Rhône-Alpes, ICNS 2012: The Eighth International Conference on Networking and Services, Grenoble, France, 2012.
- [138] Markus Döhrring and Birgit Zimmermann, "vBPMN: Event-Aware Workflow Variants by Weaving BPMN2 and Business Rules", SAP Research, Darmstadt, Germany, 2011.
- [139] Zhenyu Fang, Changqing Yin, "BPM Architecture Design Based on Cloud Computing", School of Software Engineering, Tongji University, Intelligent Information Management, Shanghai, China, 2010.
- [140] Karabagosian L, Martinez Garro J., Bazan P. "Ejecución y monitoreo de procesos de negocios distribuidos entre diferentes motores de Bonita OS" XVI Workshop de Investigadores en Ciencia de la Computación: WICC 2014. Ushuaia, Tierra del Fuego, Argentina. ISBN 978-950-34-1084-4. 2014.
- [141] NuSOAP (Agosto 2013) - <http://nusoap.sourceforge.net/>
- [142] SimpleXML (Agosto 2013) - <http://php.net/simplexml>
- [143] GraphViz (Septiembre 2013) - <http://www.graphviz.org/>
- [144] Martinez Garro J., Bazan P. "Constructing and monitoring processes in BPM using hybrid architectures" Science and Information Conference 2013, London, UK 2013.
- [145] Martinez Garro J., Bazan P. "Monitoreo de procesos distribuidos en el cloud. Una propuesta arquitectónica" WBPM 2013: Chilean Workshop on Business Process Management (BPM). Temuco, Chile. ISBN 978-956-7019-95-3. 2013.
- [146] Martinez Garro J., Bazan P. "Conceptos de dinamismo aplicados a servicios y workflows en BPMS basados en Cloud Computing" XV Workshop de Investigadores en Ciencia de la Computación: WICC 2013. Paraná, Entre Ríos, Argentina. ISBN 978-987-28179-6-1. 2013.

- [147] Patricia Bazan, Elsa Estevez, Roxana Giandini and Javier Diaz "Socializar el despliegue y seguimiento de procesos de negocios" WBPM 2014: Chilean Workshop on Business Process Management (BPM). Noviembre, 2014. Talca, Chile. 2014.
- [148] Jose Martinez Garro, Patricia Bazan and Javier Diaz "OLC y Monitoreo de procesos en el cloud: un caso de estudio" WBPM 2014: Chilean Workshop on Business Process Management (BPM). Noviembre, 2014. Talca, Chile. 2014.
- [149] Chedrese V., Parra J., Martinez Garro J., Pellitti M., Lorenzon E., Bazan P., Alvarez M. "Sistema basado en BPM para el Seguimiento del Proceso Licitatorio y la Ejecución de Proyectos del Programa PMGM-UEC-Ministerio del Interior y Transporte de la Nación Argentina" Simposio Informática en el Estado, en el marco de las 43° Jornadas Argentinas de Informática, Simposio de Informática en el Estado. ISSN 1851-2526. 2014.
- [150] Martinez Garro J., Bazan P., Diaz J. "Decomposed Processes in Cloud BPM: Techniques for Monitoring and The Use of OLC" SERP'14:2014 International Conference on Software Engineering Research and Practice <http://www.worldacademyofscience.org/worldcomp14/ws/conferences/serp14>, Las Vegas, Nevada, USA. ISBN 1-60132-286-0. 2014
- [151] Martinez Garro J., Bazan P. "Técnicas de ejecución y monitoreo de procesos en Cloud BPM" XVI Workshop de Investigadores en Ciencia de la Computación 2014 : WICC 2014. Ushuaia, Tierra del Fuego, Argentina. ISBN 978-950-34-1084-4. 2014.