

REVISIÓN DE LA LITERATURA PARA PROPONER LOS METAPROCESOS COMO ACTIVOS DE SOFTWARE

Msc. Javier D. FERNANDEZ
Ingeniería de Sistemas, Universidad de Antioquia
Medellín, Antioquia, Colombia

PhD John F. DUITAMA
Ingeniería de Sistemas, Universidad de Antioquia
Medellín, Antioquia, Colombia

RESUMEN

Este artículo tiene el propósito de mostrar algunos elementos que constituyen una propuesta de acercamiento al estado del arte como marco de referencia para un proyecto enfocado a proponer los Metaprosesos como Activos de Software. En él se muestran los antecedentes conceptuales e investigativos más relevantes que abren la discusión hacia la búsqueda de respuestas a la pregunta investigativa sobre ¿Cómo potenciar el reuso en las etapas tempranas del desarrollo de aplicaciones a partir de Metaprosesos como activos de software para dominios específicos?, la cual surge ante la evidencia de que a pesar del gran auge de sistemas de especificación de procesos de negocios o procesos de servicios de negocios, también se viene asistiendo a una serie de problemas como la falta de estándares y metamodelos de especificación en la lógica de los procesos de negocios que respondan a sus reglas y dominios específicos. Así mismo, la existencia de múltiples esquemas de trabajo y representación de Metaprosesos deja de lado el reuso planificado de activos de software en etapas tempranas del desarrollo y esquemas de formalización que posibiliten su validación para diferentes dominios.

PALABRAS CLAVES: ACTIVO DE SOFTWARE, DOMINIOS, METAPROCESO.

1. INTRODUCCION

El uso de metaprosesos como activos de software que puedan ser usados en etapas tempranas del desarrollo de aplicaciones, requiere de métodos y modelos de formalización y validación que den cuenta del proceso mismo de desarrollo al cual asisten, ya que se busca que estos se conviertan en marcos de referencia que mediante el uso de metamodelos permitan la construcción ágil de software sin apartarnos del conocimiento sobre las reglas del dominio específicos que se pretende sistematizar.

Este documento está organizado como sigue: en la sección 2 se presenta una descripción formal de los conceptos que conllevan a la contextualización del problema de investigación. En la sección 3 se presentan los antecedentes investigativos como apoyo en la investigación.

2. ANTECEDENTES CONCEPTUALES

En esta sección se darán las definiciones más relevantes que acompañan el estado del arte del proyecto de investigación en el marco de los procesos de negocios, y los metaprosesos como activos de software.

Procesos de Negocios

Un proceso de negocios como se plantea en [1], [2], [3], [4] y [5] bien se puede definir como una entidad organizacional que existe en función de la ocurrencia de eventos para cumplir un fin específico, está dirigido por reglas que permiten el control y seguimiento de las actividades que lo conforman, a estas últimas se les asignan unos recursos y roles que les ayudan a cumplir con el objetivo corporativo para el que son definidos.

Activos de Software

Según los trabajos de [6], [7] y [8] los activos de software son componentes de software que cumplen estas tres características: a) Que sean una colección de artefactos para proveer la solución a un problema b) Que puedan ser usados y reutilizados en varios contextos c) que sean extendidos y customizados en varios puntos.

Metaprosesos de Negocios

Según los trabajos de [9] los metaprosesos de negocios se pueden definir como modelos de procesos que sirven como marcos de referencia para ser instanciados en diferentes Dominios.

Modelado de Procesos de Negocios

Según [10] los procesos de negocios suelen representarse como flujos o diagramas de flujo o actividades, a veces como nodos o reglas de ejecución de tareas o actividades.

Automatización de Modelos de Procesos de Negocios

En [11] y [12] un sistema de procesos de negocios se concibe como un software de tipo genérico soportado por técnicas de diseño enfocadas al manejo de procesos de negocios operacionales, se conoce en función de los requerimientos que deba soportar el proceso y se ayuda de un esquema gráfico para su representación.

3. ANTECEDENTES INVESTIGATIVOS

En esta sección se mostraran los antecedentes investigativos más relevantes en términos de desarrollos de técnicas de modelado de procesos de negocios, técnicas de modelado de procesos de negocios para dominios específicos y técnicas de modelado de procesos de negocios para dominios específicos en el marco de uso de metaprosesos.

Técnicas de modelado de procesos de negocios

Algunas propuestas estudiadas sobre técnicas de modelado de procesos de negocios nos muestran el impacto que están han tenido en la construcción de metamodelos de negocios como marcos de representación y construcción de modelos de negocios para su uso e implementación a nivel organizacional.

[13] venían trabajando en los modelos conversacionales y de comunicación entre actores de procesos usando un nuevo paradigma llamado la perspectiva del lenguaje – acción (LAP), la cual ha sido enriquecida por la Teoría de la Acción Comunicativa. Esta perspectiva propone que una acción es ejecutada cuando alguien habla o conversa con otro en una organización. Los actos individuales de hablar se toman como requerimientos, comentarios, promesas, aceptaciones, cancelaciones, declaraciones, etc., que se consiguen a lo largo de las estructuras conversacionales que puede afectar el curso futuro de las acciones. Esta investigación sugiere que las conversaciones y las comunicaciones conducidas usando lenguajes son básicas para direccionar acciones en las organizaciones humanas. La técnica del lenguaje-acción, entonces, ve en las conversaciones para la acción una estructura clave que necesita ser analizada y modelada en el diseño de los sistemas de información y de los procesos de negocios.

[14] propone los diagramas de roles de actividades (RADs) como un conjunto de notaciones gráficas para el modelamiento de procesos de negocios. La técnica RAD particiona un proceso de negocios en un conjunto de roles interactuando. Cada una de estas actividades de roles grupales puede ser realizada por una persona, grupo o máquina (un actor o un agente). Los Roles tienen construcciones para mostrar comportamientos concurrentes o paralelos. Ellos actúan en paralelo y se comunican y sincronizan a través de interacciones. RAD es esencialmente una técnica de modelado basado en estados, acciones e interacciones de un rol que se mueve de un estado actual a un nuevo estado.

[15] propusieron un patrón de lenguaje de definición para el modelado de las transacciones del comercio electrónico, basado en varias técnicas LAP. Este patrón incluye cinco esquemas: a) primer esquema, las actuaciones son definidas como construcciones de bloques de procesos de negocios. b) una transacción es definida como la secuencia más pequeña posible de acciones que toman cierto estado. c) un Workflow es definido como un conjunto de transacciones que persiguen la realización de alguna meta. d) la noción de contrato es introducida para capturar un tipo simétrico de intercambios, en el cual todos los actores involucrados en la conversación tienen un interés común en un objeto particular. e) un escenario es descrito como la interacción que toma lugar cuando varios contratos corren concurrentemente hacia varias partes.

Otro patrón para el modelado de negocios basado en LAP fue propuesto por [16], como suplemento al patrón de [15]. Este patrón también está compuesto de cinco esquemas. Estos autores sugieren usar actos de negocios (una colección de actuaciones y materializaciones) como la unidad básica de análisis en el primer esquema, porque una interacción de negocio incluye no solo actuaciones si no también materializaciones. En el segundo esquema, usan un concepto más general – una acción par- como una transacción. Una acción par es definida como la ocurrencia de un acto de negocio funcionando como un disparador para otro acto,

el cual tendrá la función de responder. En el tercer esquema, la noción de intercambio es introducida, lo cual significa que las acciones en los negocios son siempre reciprocas. El cuarto esquema, las transacciones de negocios, se construyen en diferentes tipos de intercambios entre unos y otros. En el quinto esquema, las transacciones de los negocios entre dos actores son formuladas en un grupo de transacciones.

En resumen, las aproximaciones descritas nos muestra como las técnicas de modelado utilizan una amplia variedad de constructores simbólicos y representacionales para modelar sistemas de trabajo integrados a nivel conceptual. Mientras cada técnica utiliza un método único para proponer lo que considera son los elementos esenciales de un sistema de trabajo integrado, muchos de los conceptos usados en estas técnicas son comunes: metas, roles, actores (o agentes), actividades (o tareas o procesos), interacciones (o conversaciones), Workflow, recursos, información e interdependencia de recursos y se basan fundamentalmente en el concepto de interacción o intercambio. Ahora bien, estos elementos comunes nos brindan los componentes fundamentales para la propuesta y construcción de Metaprocesos como marcos de referencia fundamentados en el uso de modelos de procesos de negocios para instanciar en diferentes dominios.

Desarrollos de Modelos de Procesos de Negocios para Dominios Específicos

Esta sección describe como ha sido el desarrollo de los modelos de proceso de negocios o metamodelos en el contexto de los dominios específicos para la construcción de aplicaciones de software como un paso previo a su constitución como metaprocesos.

Los trabajos encontrados sobre desarrollos de modelos de negocios para dominios específicos nos remiten inicialmente a [17], quien propone un esquema de trabajo denominado “Desarrollo de Software Basado en Líneas de Productos de Software”. [17] se basa en el desarrollo de software a partir del concepto de ensamblaje de piezas de software previamente elaboradas; tal cual como sucede en los modelos convencionales de producción, cuyo eje principal es la reutilización de piezas. Dicho concepto de reutilización se fundamenta en la implementación de sistemas de software usando “activos de software” o partes de software que se pueden ensamblar en los procesos de construcción de software para diferentes dominios.

[18] proponen que la ingeniería dirigida por modelos (MDE) aparece como una gran ayuda para transferir los cambios en los procesos de negocio a los sistemas que implementan dichos procesos a través del uso de modelos. Este paradigma combina los siguientes conceptos: a) Lenguajes de dominio específico (DSL), usados para formalizar la estructura de la aplicación, el comportamiento y los requisitos dentro de un dominio particular. Los DSL son descritos usando metamodelos, que definen relaciones entre elementos dentro de un dominio. b) los Motores de transformación y generadores, los cuales analizan ciertos aspectos de los modelos que después crean varios tipos de artefactos, tales como código fuente, entradas de simulación, descripciones de uso o documentos XML, o representaciones alternativas de dicho modelo.

[19] propone una metodología que abarca desde el análisis de los requerimientos hasta el monitoreo de los procesos, apoyando las etapas de análisis, diseño, modelaje y configuración, a través del uso de patrones. La propuesta metodológica está conformada por dos macroprocesos: uno relacionado con la creación del proceso en sí mismo y otro que corresponde a la administración, y comprende: el mantenimiento, administración del proceso en producción y el monitoreo a través de indicadores de gestión. Propone una taxonomía de patrones y su representación a través de un Lenguaje de Definición de Arquitecturas (ADL) con respecto a una arquitectura de procesos, servicios y objetos canónicos, además extiende la especificación de los patrones a fin de poder medir su calidad durante el proceso de desarrollo y al producto de software obtenido a través de lo que él llama Estilos de Arquitectura Basados en Atributos (ABAS) y del uso de modelos de calidad como la norma ISO14-598.

En resumen, las técnicas descritas evidencian que las propuestas de modelado de procesos de negocios, si bien proponen esquema de trabajos comunes para el modelado de procesos de negocios independientes de las plataformas y orientadas a dominios específicos, adolecen en su mayoría de mecanismos de validación y de transformación de modelos que partan de la definición de las reglas de negocios hasta la construcción de los metamodelos de dominios específicos que aporten a la construcción de metaprocesos, esto viene acompañado de la ausencia de formalismos que propongan estrategias de validación de modelos y dominios para garantizar la consistencia del modelo con el dominio sin perder el nivel de referenciación que se pretende lograr con el uso de metaprocesos.

Desarrollos de Modelos de Procesos de Negocios para Dominios Específicos como Metaprocesos

Esta sección describe como ha sido el uso y desarrollo de los modelos de proceso de negocios o metamodelos en el contexto de los dominios específicos como propuestas de uso y configuración de metaprocesos para la construcción de software.

[20] proponen la creación de modelos de referencias a partir de la definición de los elementos comunes en un dominio específico y las variaciones que este tenga, dando lugar a modelos de procesos individualizados. La variabilidad del dominio se centra en la definición de las decisiones y sus interrelaciones, independiente de la notación usada en el modelado del proceso, asegurando con ello que los modelos de procesos resultantes sean correctamente expresados de acuerdo a las restricciones del dominio. Para capturar la variabilidad de un dominio dado se recopilan las respuestas dadas a una serie de preguntas sobre el proceso, las cuales junto con los hechos del dominio del proceso permiten configurar las variaciones del proceso en su contexto, configurando con ello el diseño del modelo. Estos hechos-proceso como puntos de variaciones de los procesos permiten configurar el dominio particular asegurando la correcta configuración semántica del modelo de dominio propuesto el cual podrá ser instanciado en otros dominios a través de los mecanismos propuestos de hechos-proceso.

[21] proponen un esquema basado en el modelado de dominios específicos con el fin de incentivar la reutilización, la articulación flexible, una mayor abstracción, la agilidad y la interoperabilidad. Fundamentados en el desarrollo de un ambiente genérico de

modelado (GME) se propone la creación de modelos de dominio específicos o (DSM) a nivel de procesos de negocios en el dominio de la aplicación y el metamodelo de entidades u objetos para la generación de código. Muestran un ciclo completo de desarrollo que parte del modelo de dominio: modelo de dominio de la empresa donde los procesos son capturados y definidos a nivel de dominio. Luego usan un GME para la generación de metamodelos tanto de procesos de negocio como de entidades para generar el modelo de entrada. Luego los metamodelos son importados en una herramienta específica. Por último, después de importar los metamodelos se aplican los elementos de la configuración y reglas de transformación para la generación de código.

[22] Proponen un esquema de valoración que pretende unificar la gestión de procesos de negocios, los sistemas de información empresariales y los estándares de modelación, exaltando los procesos de negocio como eje principal en el desempeño de los sistemas de software, y proponiendo las Redes de Petri como lenguaje de modelación que propician modelar los flujos de trabajos y almacenar datos, con los cuales se puede generar información y tomar decisiones. Concluyen en su trabajo que BPMN y BPEL son los lenguajes que más patrones de Workflow soportan y se ajustan al modelado de procesos de negocios para diferentes dominios, la incorporación en el uso de las redes de Petri como mecanismo de formalización y validación permite contribuir a su reusabilidad en otros entornos organizacionales.

[23] proponen un esquema de automatización de procesos de desarrollo de software centrado en el uso del Software Process Engineering Metamodel (SPEM) con el fin de utilizar motores Workflow para automatizar procesos de negocio. Para lo cual se deberá definir una transformación del metamodelo SPEM al metamodelo de Business Process Modeling Notation (BPMN) por medio del lenguaje Relations que forma parte de Query/Views/Transformations (QVT). La especificación BPMN resultante podrá ser transformada a un lenguaje estándar para la implementación de procesos Workflow, como puede ser Business Process Execution Language for Web Services (BPEL4WS) o XML Process Definition Language (XPDL). Con esto se lograría fundamentalmente la automatización de cualquier proceso de desarrollo de software especificados bajo el SPEM a través de su transformación a procesos Workflow de tipo estándar para su uso como metamodelo. Esta aproximación se centra en el concepto de "metodologías de desarrollo", encargadas de guiar la producción de software. Concluyen que el esquema general de la transformación de procesos de desarrollo de software basados en SPEM a Workflow puede ser visto en tres niveles: Metamodelo, Definición/Modelo y Ejecución. En el primer caso se construye un modelo del proceso de negocio, luego este se transforma en un metamodelo que será traducido a un tipo de Workflow específico para ser ejecutado como modelo de negocio y como aplicación de software específica.

[24] proponen la articulación de los BPMS con el Desarrollo Dirigido por Modelos para resolver las cuestiones de integración de servicios en un proceso de negocio. Parten de reconocer la carencia de casos de estudio sobre BPM y una gran confusión sobre qué es y qué no es un BPMS. Para la integración de servicios en el proceso de negocio muestra una aproximación basada en transformaciones de modelos.

El esquema del proceso de desarrollo propuesto es el siguiente: Desarrollar las interfaces requeridas, tanto del sistema (internas y externas) como humanas. Generar un modelo BPMN inicial a partir de una coreografía WS-CDL que dé cuenta de la interacción humano-aplicación. Diseñar el modelo BPMN. Generar el código BPEL en base al modelo BPMN. Desplegar el código BPEL y las interfaces en el motor de ejecución del BPMS. Monitorizar los procesos en ejecución mediante las interfaces de administración y monitorización.

[25] propone el desarrollo de software con enfoque en el negocio, el cual se centra en conjuntar la visión del negocio centrada en especificar y mejorar sus procesos mediante análisis del negocio, y la visión de TI centrada en informatizar dichos procesos evolucionando en la tecnología y metodologías de desarrollo de software. Plantea que el enfoque de diseño Service Oriented Architecture (SOA) promete cumplir este desafío conjuntando el enfoque de Business Process Modeling (BPM) con el desarrollo orientado a servicios, así mismo, plantea que el enfoque de desarrollo Model Driven Architecture (MDA) propone aportes a la automatización del desarrollo que permite hacer este tránsito entre el enfoque de diseño orientado al servicio y el enfoque del diseño orientado al proceso. Reconoce la necesidad de enfocar el desarrollo de software en los procesos del negocio de la organización con el fin de establecer paradigmas de desarrollo o construcción de software centrados en los procesos de negocios y los servicios, lo cual constituye un insumo fundamental a la definición de los metaprosesos.

Metodologías de Desarrollo de Software Orientados por Procesos

En [26] se presentan una serie de metodologías revisadas que soportan la producción de software orientada a procesos, estas son: EPOS: Expert System for Program and System Development o Modelado de Procesos Cooperativos Orientados a Objetos [27], SOCCA: Specifications of Coordinated and Cooperative Activities [28], MERLIN: Supporting Cooperation in Software Development Through a Knowledge-Based Environment [29], OIKOS: Constructing Process-Centred SDEs [30], ALF: A Framework for Building Process – Centred Software Engineering Environments [31], ADELE-TEMPO: An Environment to Support Process Modeling and Enaction [32], SPADE: An Environment for Software Process Analysis, Design, and Enactment [33], PEACE: Goal –Oriented Logic – Based Formalism for Process Modeling [34], E³: Object – Oriented Software Process Model Design [35], PADM: Towards a Tool Process Modeling System [36],

En una segunda instancia se compararon las anteriores metodologías, recogiendo lo trabajado en [26]:

| FACTOR DE EVALUACION | EPOS | E ³ | MERLIN | PWI | SPADE | SOCCA | OIKOS | ALF | PEACE | ADELE |
|--|------|----------------|--------|-----|-------|-------|-------|-----|-------|-------|
| 1. PROPOSITO: PROPOSITO PRINCIPAL EN EL MODELADO DE PROCESOS DE SOFTWARE | 8 | 7 | 13 | 12 | 9 | 8 | 7 | 6 | 9 | 6 |
| 2. TIPOS DE COMPONENTE BASICOS: ITEMS PRINCIPALES QUE PERMITEN EL MODELAMIENTO DE LOS PROCESOS DE SOFTWARE | 11 | 10 | 11 | 12 | 12 | 11 | 9 | 10 | 12 | 12 |
| 3. MODELAMIENTO DE LOS TIPOS DE COMPONENTES BASICOS | 16 | 13 | 10 | 10 | 12 | 9 | 7 | 11 | 9 | 15 |
| 4. LENGUAJES | 10 | 2 | 4 | 4 | 6 | 8 | 4 | 2 | 4 | 6 |
| 5. PROPIEDADES DEL LENGUAJE DE MODELADO DE PROCESOS | 12 | 8 | 8 | 6 | 9 | 12 | 10 | 10 | 11 | 4 |
| 6. HERRAMIENTAS DE SOPORTE PARA ESCRIBIR LOS MODELOS DE PROCESOS DE SOFTWARE | 16 | 12 | 2 | 12 | 15 | 2 | 8 | 12 | 10 | 10 |
| 7. METAPROCESO: ANALISIS Y DISEÑO | 4 | 4 | 0 | 4 | 4 | 2 | 2 | 4 | 4 | 2 |
| 8. CUSTOMIZACION, INSTANCIACION Y EVOLUCION | 10 | 10 | 4 | 12 | 10 | 0 | 6 | 10 | 12 | 12 |
| 9. SOPORTE PARA LA REPRESENTACION | 26 | 22 | 14 | 20 | 24 | 0 | 18 | 23 | 24 | 21 |
| 10. REPRESENTACION INTERNA | 4 | 4 | 4 | 4 | 4 | 0 | 4 | 4 | 4 | 4 |
| 11. ARQUITECTURA DEL MOTOR DE PROCESOS | 2 | 2 | 2 | 2 | 2 | 0 | 2 | 2 | 2 | 2 |
| TOTAL | 119 | 94 | 72 | 98 | 107 | 52 | 77 | 94 | 101 | 94 |

TABLA 1. Comparación de metodologías orientadas a los procesos de software

Gráfico Comparativo de Metodologías Orientadas a Procesos de Software

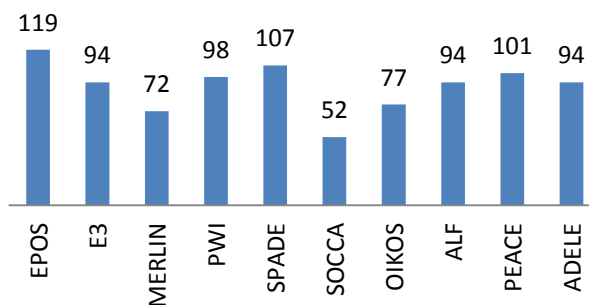


FIGURA 1. Resultados de la Comparación de Metodologías

Dicha comparación se logró estructurar a partir de los resultados presentados por [37], el cual se estructuró en cuatro partes fundamentalmente: Acercamiento al modelado de procesos de

software, lenguajes de modelado de procesos de software, metaprocesos y motores de representación. Para efectos de esta tesis se siguió la metodología de calificar de (0=No Presenta Información, 1= Presenta Parcialmente, 2=Presente Totalmente), para efectos de cualificar y poder establecer un acercamiento a la mejor metodología que cumpla con los criterios perseguidos en el desarrollo de este proyecto.

Se evaluaron once (11) factores, cada uno con sus respectivos parámetros; estos factores obedecen al esquema presentado en [37] a partir de una serie de preguntas que permitiera inferir el parámetro evaluado. Dichos factores fueron: 1. Propósito ¿Cuál es el propósito principal en el modelado de procesos de software?, 2. Tipos Básicos de Componentes ¿Cuáles son los ítems principales para el modelamiento de procesos de software?, 3. Modelamiento de los tipos Básicos de Componentes ¿Cuál paradigma/modelo o técnica ha sido adoptada como una base conceptual para la descripción de los tipos de componentes principales?, 4. Lenguajes ¿Cuáles son los paradigmas que gobiernan el diseño y la definición de los lenguajes de modelado de procesos?, 5. Propiedades del Lenguaje de modelado de procesos ¿Cuáles son las propiedades principales del Lenguaje de modelado de procesos?, 6. Herramientas que soportan la escritura de los modelos de procesos de software ¿Cuáles herramientas son provistas para escribir los modelos?, 7. Análisis y diseño del Modelado de Procesos de software o Metaproceso ¿Se tienen metacatividades y cuáles son los resultados?, 8. Customización, instanciación y evolución ¿Cómo es soportada la customización, instanciación y evolución?, 9. Soporte para la representación ¿Cuáles elementos básicos y mecanismos son direccionados para soportar la representación?, 10. Representaciones internas ¿Son descritos los SPMs usando una representación específica interna para el motor de representación?, 11. Arquitectura del motor de procesos ¿Es el motor de procesos único o hay posibilidad de varios motores?

Una vez revisadas las valoraciones se puede concluir que las metodologías EPOS, PWI y PEACE, son las más completas en cuanto al cumplimiento de los factores evaluados en su respectivo orden. Así mismo, sobresale el hecho de cada parámetro se muestra de manera representativa en el dominio de las metodologías evaluadas por factores, siendo esto que para el factor Propósito sobresale el parámetro Aplicación como el predominante, en el factor Tipos Básicos de Componentes sobresale los agentes humanos y computarizados, en cuanto al Modelamiento de los tipos Básicos de Componentes es representativo el modelado entidad-relación, en cuanto a Lenguajes sobresalen la formalización y la orientación a objetos, sobre Propiedades del Lenguaje de modelado de procesos sobresalen representatividad, interpretabilidad y compilabilidad, sobre las Herramientas que soportan la escritura de los modelos de procesos de software sobresale el soporte sintético, en cuanto al Análisis y diseño del Modelado de Procesos de software o Metaproceso se tiene el método como elemento fundamental del Metaproceso, en cuanto a la Customización, instanciación y evolución son preponderantes los mecanismos para la customización y la instanciación, sobre el Soporte para la representación sobresale el determinismo, preservación de la consistencia, invocación de herramientas y consistencia de los objetos, en cuanto a las Representaciones internas no sobresale ningún parámetro en específico siendo preponderante tanto la representación interna del SPM como de su estado representado y finalmente en cuanto a la Arquitectura del motor de procesos sobresale la existencia de varios motores.

Visto lo anterior, se hace necesario que la propuesta a construir sobre metodologías o metaprocesos como activos de software recojan los resultados anteriores, en términos de que este debe cumplir principalmente mas no excluyentemente con los parámetros más relevantes presentados en las metodologías estudiadas, es decir, enfocado en su aplicación y a la interacción con los agentes humanos y computarizados, que considere las relaciones y las entidades existentes, fundamentalmente que tenga un componente formal y orientado a objetos, representativo o expresivo, fácil de interpretar y que se pueda compilar, además, que permita una definición sintáctica de los modelos, que presente un método, que permita a través de sus conceptos y mecanismos ayudar a ir de un modelo genérico a un modelo adaptado para un dominio específico y que presente metaactividades que permitan su instanciación, además, deberá ser fundamentalmente determinístico, que no se deje afectar por la intervención de los agentes humanos, preserve la consistencia y permita identificar inconsistencias fácilmente, que permita el uso de herramientas externas para su integración y mantenga la consistencia entre los objetos del modelo, así mismo que permita la representación interna del modelado del proceso de software y que permita el uso de varios motores para el manejo de los procesos.

4. AGRADECIMIENTOS

Al proyecto de Telesalud del Centro de Excelencia ARTICA, a Colciencias y el Ministerio de Tecnologías de la Información y las Comunicaciones de Colombia.

5. REFERENCIAS

- [1] Alfaro, Juan José. Sistemas para la medición del rendimiento en la empresa, México, Limusa, 2008.
- [2] Hammer, M. Champy, J; Reengineering the corporation: A manifesto for business revolution. New York, Mc Graw Hill, 1993.
- [3] Hammer, M. Zisman, M; Design and implementation of office information systems. In Proceedings NYU Symposium on automated office systems, New York University graduate school of business administration, pages 13-24. 1979.
- [4] Davenport, T, H. Reengineering a business process. Harvard business school note, 9-396-054, pages 1-16, USA, 1995.
- [5] Kettinger, J. Teng, C., Guha, S. Business Process change: a study of methodologies, techniques and tools, MIS Quarterly, 21 (1), pages 55-80, USA, 1997.
- [6] Larsen, G. Model-Driven development: Assets and reuse, USA, IBM Systems journal, 45(3), 2006.
- [7] O. Ávila-García, A. Estévez García, V. Sánchez Rebull, and J. L. Roda García. Using software product lines to manage model families in model-driven engineering. In SAC 2007: Proceedings of the 2007 ACM Symposium on Applied Computing, track on Model Transformation, pages 1006_1011. ACM Press, Mar 2007.
- [8] P. Clements and L. Northrop. Software Product Lines: Practices and Patterns. Addison Wesley, Aug 2001.

- [9] Rolland, C., Prakash N. On the Adequate Modeling of Business Process Families. Université Paris1 Panthéon Sorbonne. Francia, 2000.
- [10] Basu, A. y Blanning, R. Model integration using metagraphs. *Information systems research*, USA, 5(3), 1994.
- [11] Carlsen, S. Conceptual modeling and composition of flexible workflow models. PhD Thesis, Information system group. Department of computer and information science., Faculty of physic, informatics and mathematics, Norwegian university of science and technology, Trondheim. 1997.
- [12] Davenport, T. H. & J. Short. The New Industrial Engineering: Information Technology and Business Process Redesign, *Sloan Management Review* 31(4), 11-27. 1990.
- [13] Winograd, T.; Flores, F. Understanding Computers and Cognition. A New foundation for Design. Norwood (NJ). 1986.
- [14] Ould, M.A. Business Processes: Modeling and Analysis for Re-engineering and Improvement, Wiley, Chichester, 1995.
- [15] Weigand, H., Verharen E., & Dignum F. (1996). Interoperable transactions in business models - a structured approach. University of Technology, Eindhoven. 1-17. Disponible en: <http://www.cs.uu.nl/~dignum/papers/caise96.pdf> consultado el 10-Agosto-2009.
- [16] Lind, M., Goldkuhl, G., (2005) The Evolution of a Business Process Theory - the Case of a Multi-Grouped Theory, in Beekhuizen J, von Hellens L, Guest K, Morley Understanding Computers and Cognition. A New foundation for Design. Norwood (NJ). 1986.
- [17] Montilva, J. Desarrollo de Software Basado en Líneas de Productos de Software. IEEE Computer Society. Mérida. Venezuela, 2006.
- [18] Perez, J. Ruiz, F. Piattini, M. Model Driven Engineering applicator a Business Process Management. Informe Técnico UCLM-TSI-002. Departamento de Tecnologías y Sistemas de Información. Universidad de Castilla-La Mancha. España, 2007.
- [19] Bonillo, P. Metodología para la Gerencia de los Procesos del Negocio Sustentada en el Uso de Patrones. *Journal of Information Systems and Technology Management*. Vol. 3, No. 2, 2006, p. 143-162, 2006.
- [20] La Rosa, M. et al. Linking Domain Models and Process Models for Reference Model Configuration. *Lecture Notes in Computer Science*. Vol. 4928, Springer, Berlin, p. 417-430, 2008.
- [21] Mohan, S. Choil, E. Dugki, M. Domain Specific Modeling of Business Processes and Entity Mapping using Generic Modeling Environment (GME). *International Conference on Convergence and Hybrid Information Technology*. Vol. 1, p. 533-538, 2008.
- [22] Valdivia, M.; Santana, W. Valoración teórica sobre la gestión de procesos de negocios, los sistemas de información empresariales y los estándares para la modelación, UCI, Cuba. Disponible en: http://semanatecnologica.fordes.co.cu/Evirtual/files/Marby%20%20William_VALORACION%20TEORICA%20SOBRE%20LOS%20SISTEMAS%20DE%20INFORMACION%20EMPRESARIALE%20S%20Y%20LOS%20ESTANDARES%20PARA%20LA%20MOD%20ELACION.doc. Consultado 10-Diciembre-2009. 2008.
- [23] Zorzan F.; Riesco D. Automatización de procesos de desarrollo de software definidos con SPEM. Universidad de Rio Cuarto, Argentina. Disponible en: <http://www.ing.unp.edu.ar/wicc2007/trabajos/ISBD/070.pdf>. Consultado 10-Diciembre-2009. 2008.
- [24] Cánovas et al. Un caso de estudio para la adopción de un BPMS. Universidad de Murcia. España. Disponible en: <http://alarcos.inf-cr.uclm.es/pnis/articulos/pnis-07-canovas-bpms.pdf>. Consultado 10-Diciembre-2009. 2008.
- [25] Delgado, A. Desarrollo de Software con enfoque en el Negocio, Universidad de la Republica, Uruguay. Disponible en <http://alarcos.inf-cr.uclm.es/pnis/articulos/pnis-07-delgado-dsen.pdf>. Consultado 10-Diciembre-2009. 2008.
- [26] Finkelstein, A. et al. Software Process Modeling and Technology, Research Studies Press LTD. Londres. 1994.
- [27] Conraid, R. et al. Object –Oriented and Cooperative Process Modelling in EPOS. In PROMOTER Book. Ed. Bashar A. Nuseibeh, Imperial College. 1994.
- [28] Engels, G. et al. Object –Oriented Specification of Coordinated Collaboration. In *Advanced IT tools: IFIP World Conference on IT Tools*, Springer. Canberra (Australia). 2-6 September, 1996.
- [29] Junkermann, G. et al. Merlin: Supporting Cooperation in Software Development through a Knowledge-based Environment. In *Software Process Modelling Technology*, A. Finkelstein, J. Kramer, & B. Nuseibeh, eds., ch. 5, pp. 103-130, Somerset, England: Research Studies Press, 1994.
- [30] Montangero, C. & Ambriola. V. OIKOS: Constructing Process-Centered SDEs. In A. Finkelstein, J. Kramer, & B. Nuseibeh, editors, *Software Process Modelling and Technology*, pages 33 – 70. John Wiley & Sons Inc., 1994.
- [31] Canals G. et al. ALF: A framework for building process-centred software engineering environments. In A. Finkelstein, J. Kramer, & B. Nuseibeh, editors, *Software Process Modelling and Technology*, pages 153 – 185. John Wiley & Sons Inc., 1994.
- [32] Belkhatir N. et al. ADELE-TEMPO: An environment to support process modeling and enactment. In A. Finkelstein, J. Kramer, and B. Nuseibeh, editors, *Software Process Modelling and Technology*, pages 187 – 222. John Wiley & Sons Inc., 1994.
- [33] Bandinelli, S. et al. SPADE: An Environment for Software Process Analysis, Design, and Enactment. In A. Finkelstein, J. Kramer, and B. Nuseibeh, editors, *Software Process Modelling and Technology*. John Wiley & Sons Inc., 1994.
- [34] Arbaoui, S. & Oquendo, F. PEACE: goal-oriented logic-based formalism for process modelling. In A. Finkelstein, J. Kramer, and

B. Nuseibeh, editors, Software Process Modelling and Technology. John Wiley & Sons Inc., 1994.

[35] Baldi, M. Et al. E3: object-oriented software process model design. In A. Finkelstein, J. Kramer, and B. Nuseibeh, editors, Software Process Modelling and Technology. John Wiley & Sons Inc., 1994.

[36] Bruynooghe, F. et al. PADM: towards a total process modelling system. In A. Finkelstein, J. Kramer, and B. Nuseibeh, editors, Software Process Modelling and Technology. John Wiley & Sons Inc., 1994.

[37] Lonchamp, J. An assessment exercise. In A. Finkelstein, J. Kramer, and B. Nuseibeh, editors, Software Process Modelling and Technology. John Wiley & Sons Inc., 1994.