

Arquitectura para la Coordinación de Flujos de Trabajo de MoProSoft por Niveles de Capacidad de Procesos

Leonel Valenzuela Ruiz¹, Brenda Leticia Flores Rios², José Martín Olguín Espinoza³

Resumen—Con la existencia de un modelo de procesos para la industria mexicana de software se generó la necesidad de contar con herramientas para su implantación. En este documento se expone la metodología utilizada para la trazabilidad entre MoProSoft por niveles de capacidad de procesos y el modelo de requisitos de una arquitectura base para la interpretación e implantación de dicho modelo de procesos. Se presenta el diseño arquitectónico de la herramienta fTIMoN para la coordinación de los diagramas de actividades definidos en MoProSoft.

Palabras Clave—Arquitectura, Modelo de Procesos, MoProSoft.

I. INTRODUCCIÓN

EN México, las Micro, Pequeñas y Medianas empresas (MiPyMES) emplean el 78% del total de la población económicamente activa, las cuales se estratifican en los sectores industrial, comercio y de servicios [1]. A partir de que la Secretaría de Economía definió, en el 2001, el Programa para el Desarrollo de la Industria del Software (PROSOFT), las MiPyMES orientadas a la industria de software se han agrupado en Clusters como un medio para elevar y extender la competitividad del país, mediante la estrategia de promover el uso y aprovechamiento de las Tecnologías de Información (TI) [2], intercambiar experiencias y conocimientos e incrementar la capacidad de madurez de procesos de desarrollo de software. El Distrito Federal, Nuevo León, Estado de México, Jalisco y Baja California fueron las primeras entidades federativas que mostraron acciones sobre inversión en proyectos productivos, un impulso en la creación de parques tecnológicos y la promoción de una figura de empresas integradoras [2]. En el 2004, se anunció formalmente la constitución del Cluster de Tecnologías de Información y Software de Baja California, A. C. (IT@Baja) [3]. Hasta el

momento, existen 21 Clusters de TI conformados a nivel nacional y se estima que para finales del 2006, se hayan creado otros 8 [2]; lo que representará que casi todos los estados de la república mexicana contarán con Clusters relacionados con las TI.

Paralelamente a dichos sucesos, el comité de PROSOFT analizó que la industria mexicana de software requería establecer una disciplina y cultura de procesos para mejorar el desarrollo de software y obtener resultados exitosos y predecibles. Dicho comité evaluó la adopción de los modelos ISO 9000, ISO 15504 y SW-CMM, pero el resultado determinó que ninguno de los estándares o modelos cumplía con los requisitos expresados por la industria mexicana de software, por lo que se determinó la necesidad de contar con un modelo de procesos nacional basado en las mejores prácticas disponibles y reconocidas a nivel mundial y mecanismos de evaluación que correspondieran al modelo identificado [4]. MoProSoft, es el modelo de procesos orientado a la industria mexicana de software con el objetivo de incorporar las mejores prácticas en Gestión e Ingeniería de Software, permitiéndole a las MiPyMES elevar su capacidad de ofrecer servicios y productos de software con calidad [5].

MoProSoft está dirigido a las MiPyMES o áreas internas dedicadas al desarrollo y/o mantenimiento de software. Las empresas que no cuenten con un modelo de procesos pueden adoptarlo ajustándolo a sus necesidades, y las que ya cuenten con algunos procesos definidos pueden usarlo como punto de referencia. MoProSoft y su método de evaluación EvalProSoft fueron publicados en el Diario Oficial de la Federación el 15 de Agosto de 2005 como una Norma Mexicana (NMX)[6]. MoProSoft versión 1.3 (versión coloreada) indica con 4 colores el contenido del modelo dependiendo el nivel de madurez al que pertenece o desea pertenecer la organización [7]. Es decir, MoProSoft versión 1.3 se encuentra estructurado por niveles de capacidad de procesos.

Este documento se estructura de la siguiente forma: en el apartado 2 se comenta brevemente la problemática asociada a la carencia de herramientas que apoyen la interpretación o implantación de MoProSoft; en la sección 3 se presenta la metodología que se utilizó para obtener la arquitectura de la aplicación propuesta, misma se detalla en el apartado 4; finalmente, en la sección 5 se exponen las conclusiones.

El desarrollo de este trabajo de investigación fue realizado con el apoyo económico proporcionado por el Consejo Nacional de Ciencia y Tecnología (CONACYT).

¹ Estudiante del programa de Maestría y Doctorado en Ciencias e Ingeniería (MYDCI), en el Instituto de Ingeniería de la Universidad Autónoma de Baja California (leonel_vr@uabc.mx).

² Jefa del Departamento de Computación e Informática, en el Instituto de Ingeniería de la Universidad Autónoma de Baja California (bflores@uabc.mx)

³ Profesor Titular de la carrera de Ingeniero en Computación, en la Facultad de Ingeniería de la Universidad Autónoma de Baja California (molguin@uabc.mx).

II. CONTEXTO DE LA PROBLEMÁTICA

Desde Octubre de 2005, los empresarios y público interesado disponen del modelo MoProSoft en sus versiones impresas 1.1 y 1.3. Ese mismo año, se pusieron a disposición las aplicaciones Kualí y MDM para el soporte en la implantación de MoProSoft:

--**Kualí**, una herramienta proporcionada por la Secretaría de Economía como parte del proyecto Acelera.Prosoft [8]. Kualí maneja como elementos de trabajo a los defectos, riesgos, requerimientos y tareas. Además, cuenta con la definición de catálogos por contactos, roles, usuarios, módulos y compañías, entre otras características (Tabla I).

--**Manejador de Documentos MoProSoft (MDM)**, desarrollada en la Universidad de las Américas como una herramienta disponible en Internet, que cual permitiera a los usuarios utilizar plantillas para documentar los procesos de Gestión de Negocio, Gestión de Procesos y Administración de Proyectos Específicos definidos en MoProSoft [4] (Tabla I).

TABLE I
CARACTERÍSTICAS DE LAS HERRAMIENTAS KUALI Y MDM.

Kuali	MDM
Consulta de la documentación del modelo manejando todos los roles	Consulta de la documentación de 3 procesos
Incurción de elementos de trabajo como: documentos, defectos, requerimientos, riesgos, y tareas.	
Todos los elementos de trabajo, roles e información en general se almacena localmente	Los elementos de trabajo se almacenan en una base de datos centralizada
Dependencia total del sitio de Kualí para acceder a la aplicación.	Dependencia del servidor WEB
No incluye diagramas de actividad.	

La tecnología necesaria para la instalación y uso de Kualí y MDM se presentan en la Tabla II.

Las aplicaciones MDM y Kualí no cubren por completo el patrón de procesos definido en MoProSoft versión 1.3. MDM solo incorpora 3 procesos limitando al usuario en el seguimiento, interrelación e implantación de los demás procesos. Kualí, por su parte al ser una versión Beta, está en constante modificación y actualización en tiempo real. Además, requiere de varios componentes para su instalación (Tabla I), no se dispone de un manual de usuario y cuenta con una dependencia al sistema operativo Windows.

Esto lleva a la necesidad de fortalecer e integrar esfuerzos para hacer instancias más concretas para los procesos MoProSoft con las prácticas y metodologías de las MiPyMEs, generar herramientas de apoyo a MoProSoft basadas en estándares y contar con un sitio Web MoProSoft

para el intercambio de prácticas, plantillas y experiencias que se vayan generando a partir de la interpretación e implantación del modelo [9]. Por otro lado, debido a que se han presentado iniciativas de estandarización de lenguajes de procesos como son XPDL, BPML, ebXML-BPSS o BPEL4WS [10] se requiere de la integración de la Ingeniería de Procesos e Ingeniería de Software que utilicen tecnologías Web y el Lenguaje de Marcado eXtensible (XML). La importancia de generar herramientas de apoyo a MoProSoft tomando en cuenta un patrón de procesos, estándares orientados al contexto y/o tecnología WEB permitirá cubrir las necesidades expuestas por [9 y 10].

TABLE II.
TECNOLOGÍA REQUERIDA POR LAS APLICACIONES KUALI Y MDM.

Kuali	MDM
Windows Installer 3.1	Tomcat Server
SQL Server Express	MySQL
.NET Framework 2.0	Java
Crystal Reports para .NET 2.0	HTML/Flash
Microsoft Data Access / Components (2.8) Win	

Por lo anterior, se formuló como objetivo general el diseñar un ambiente para la coordinación de los flujos de trabajo definidos en MoProSoft por niveles de capacidad de procesos para que las MiPyMEs, el sector de gobierno y la academia orientadas a la industria de software lo puedan utilizar como soporte en la implantación de dicho modelo.

III. METODOLOGÍA

La metodología utilizada para obtener el diseño arquitectónico del ambiente deseado para la coordinación de flujos de trabajos definidos en MoProSoft es la propuesta por García Molina *et al.* [11] (Figura 1).

El modelo de procesos es representado a través del diagrama de actividades y el catálogo de procesos porque ambos se especifican dentro del patrón de procesos de MoProSoft y definen la jerarquía de cada proceso, los artefactos, actividades, roles y reglas involucrados. Un enfoque similar es utilizado por Eriksson y Penker al definir un patrón del negocio (Business Goal Decomposition) para ser utilizado como un modelo de objetivo guía para la descomposición de los objetivos de la organización [12]. Una vez analizado el modelo de procesos es necesario realizar un análisis de mercado y técnico sobre la tecnología y estándares existentes para conocer sus ventajas y desventajas.

En el análisis de requisitos se definen los requerimientos funcionales y no funcionales identificados, las especificaciones del sistema y se genera su documento de requerimientos para proveer las bases para el diseño del software. Por último, en el modelo de requisitos se requiere obtener un diseño de bajo nivel [11].

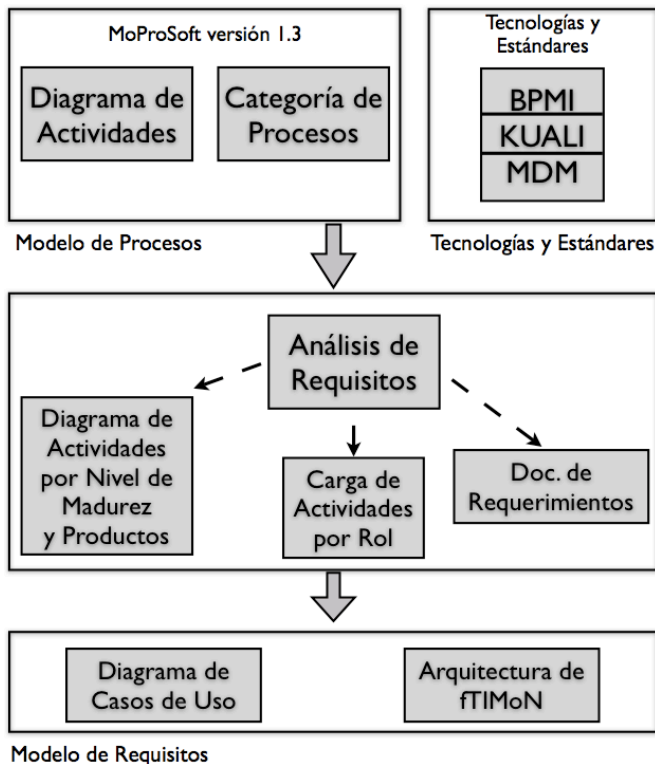


Figura 1. Relación de trazabilidad entre el modelo de procesos y el modelo de requisitos.

A. Identificar procesos y roles MoProSoft

Experiencias 2005 y 2006 en Baja California

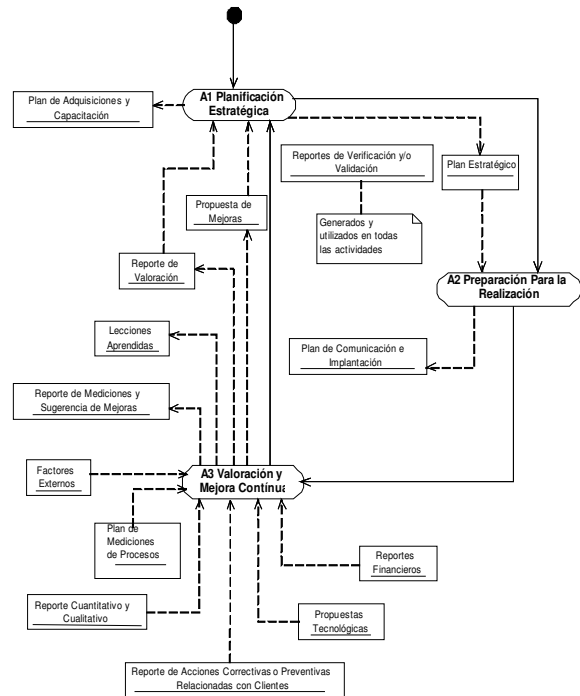
Siendo Baja California uno de los estados en donde se ha presentado una vinculación significativa por parte del sector empresarial, académico y de gobierno; el primer paso consistió en analizar el patrón de procesos de MoProSoft. En Junio de 2005 se organizó la Semana de MoProSoft con participantes de las ciudades de Ensenada, Tijuana y Mexicali y con sede en la Universidad Autónoma de Baja California, campus Mexicali. Dicho evento fue el primero en su tipo, al no haberse dado antes un taller de interpretación de MoProSoft [9].

El ejercicio se continuó en Abril de 2006, con el taller Interpretación para la implantación del modelo MoProSoft, impartido por la compañía ItEra. Dos meses más tarde y contando con la participación de los mismos sectores de las 3 ciudades mencionadas, se llevó a cabo la segunda Semana MoProSoft, dirigido nuevamente por la directora del proyecto MoProSoft. Esta primera etapa ha permitido la conceptualización del patrón de procesos utilizado y un entendimiento de los artefactos y técnicas requeridas en la implantación de los procesos definidos.

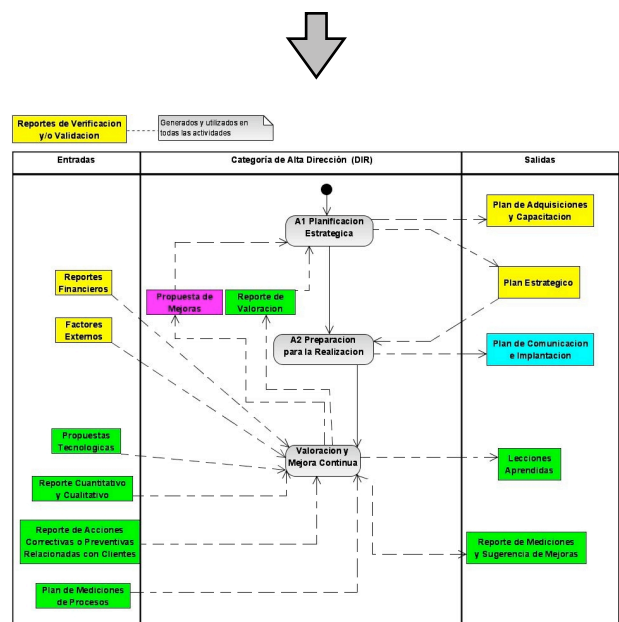
Análisis de procesos MoProSoft

Paralelamente al desarrollo de la capacitación, se inició con un análisis de los artefactos y roles involucrados entre tales actividades. Un aspecto notable entre MoProSoft versión 1.1 a 1.3 fue la especificación de las actividades por nivel de madurez dentro de la documentación del patrón de procesos, pero los diagramas de actividades no presentaron

ningún cambio; por lo que se consideró hacer una propuesta de reestructuración de diagramas de actividades tomando en cuenta además del nivel de madurez los productos de entrada y salida (Figura 2). El objetivo es facilitar gráficamente la identificación e interpretación de los productos necesarios para desempeñar el proceso y los productos generados durante el mismo, dependiendo el nivel en el que se encuentre la organización.



a) Ejemplo de diagrama de actividades de MoProSoft versión 1.3



b) Propuesta de reestructuración de diagramas de actividades

Figura 2. a) Diagrama de actividades de MoProSoft, b) Diagrama que toma en cuenta el nivel de madurez y los productos de entrada, salida e internos.

Una vez identificados los procesos, es preciso encontrar los roles involucrados en su realización. Cada uno de los agentes o actores de la organización desempeñan cierto papel (rol) cuando colaboran con otros para llevar a cabo las actividades [10]. En el caso de MoProSoft ya se identifican los roles autoridades y responsables para cada proceso ya sea como agentes propios de la organización o externos (como los clientes). El rol responsabilidad es el principal responsable por la ejecución del proceso, mientras que el rol autoridad es el rol responsable por validar la ejecución del proceso y el cumplimiento de su propósito [7]. También existen otros roles involucrados quienes cuentan con ciertas habilidades para la ejecución de actividades o tareas específicas. Por esta razón, se propone un diagrama de carga de actividades de roles (Figura 3). Dicho diagrama podrá visualizar gráficamente el número de roles de un proceso específico con el número de actividades en las que está involucrado. De esta manera, se podría observar aún más la carga de actividades del rol responsable y del rol autoridad contra los demás roles involucrados, quienes pueden realizar al menos una tarea dentro de todo el proceso (Figura 3). Con el diagrama propuesto podría darse a conocer a los roles en cuál(es) actividad (actividades) está involucrado, su importancia al realizar las tareas de dicha(s) actividad (actividades) y su interacción con otros roles.

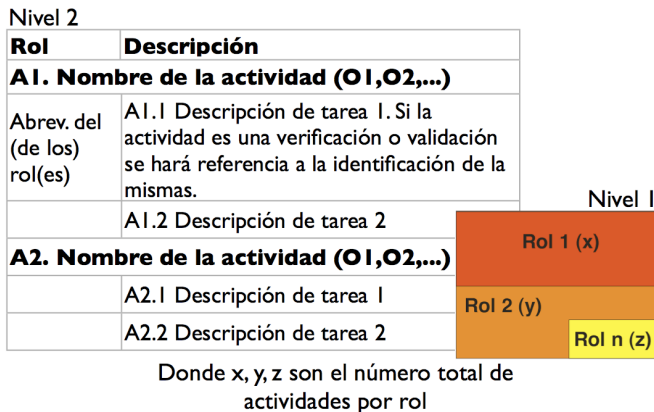


Figura 3. Diagrama de carga de actividades de roles por proceso.

B. Analizar tecnologías y estándares existentes

--Business Process Management Initiative (BPMI).

En el contexto de procesos, BPMI es un organismo responsable de la creación de estándares [13] y proporciona una estructura de procesos, en la que se definen actividades, entradas, salidas, roles y demás elementos de acuerdo a las reglas de la organización (Figura 4). De esta manera, se asegura que las tareas sean asignadas correctamente a los roles involucrados y ejecutadas en el orden establecido, obteniendo la información requerida por el proceso[14].

El componente BPMN (Business Process Modeling Notation) provee entre otras cosas una notación gráfica para expresar los procesos organizacionales en los diagramas de procesos, permite la implementación de procesos basados en el lenguaje de marcado XML. BPSM (Business Process Semantic Model) define los procesos en metamodelos y utiliza lenguajes orientados a dichos metamodelos.

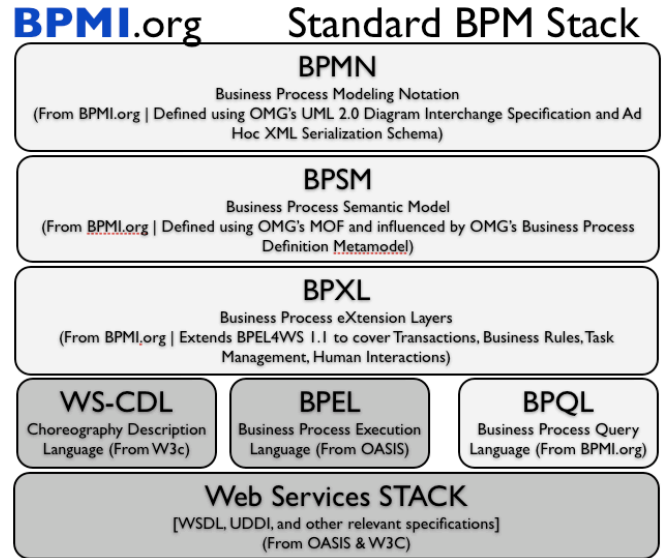


Figura 4. Modelo BPMI.

La coordinación asíncrona entre servicios es coordinada por el componente BPEL (Business Process Execution Language) con la lectura de un archivo definido con un lenguaje de marcado.

El estándar BPXL es una extensión de BPEL, define la interacción con los usuarios humanos, identifica roles y relaciones entre personas. WS-CDL se encarga de la colaboración punto a punto de los servicios WEB, así como de la coreografía entre estos. El componente BPQL provee un servidor y repositorio de procesos, y permite la consulta para el análisis y control de ejecución.

--**WebSphere** ®. Es una aplicación de IBM que provee la funcionalidad de modelar gráficamente procesos de negocios con la simbología del Lenguaje Unificado de Modelado (UML). Define roles, productos de entrada y salida, tiempo de ejecución de tareas entre otros.

--**Oracle BPEL Process Manager 2.0**. Plugin de Eclipse que permite modelar flujos de trabajo, productos de entrada y salida generando su respectivo archivo BPEL representado por medio de un lenguaje de marcado.

--**DBXML**. Manejador de colecciones compuesto por archivos de tipo XML.

En base al análisis de tecnologías y estándares descritos anteriormente, se eligió la herramienta WebSphere para modelar los diagramas de flujo de trabajo utilizando el patrón de procesos de MoProSoft para posteriormente generar su respectivo BPEL. Este será almacenado en un repositorio de procesos permitiendo ser consultado y ejecutado por el motor de procesos. Se descartó Oracle BPEL Process Manager 2.0 por no contar con una opción para la definición de roles y no utilizar la simbología de UML, relevante para la representación de los diagramas de actividades de MoProSoft. Por otro lado, la consulta y almacenamiento de la base de conocimiento y lecciones aprendidas se podrá realizar por medio del manejador DBMXL.

No se tomaron en cuenta los componentes BPMN y BPSM, especificados dentro del modelo de la Figura 4,

debido a que estos proveen funcionalidad para modelar procesos gráficamente y crear su representación en metamodelos. MoProSoft ya define procesos y flujos de trabajo por medio del patrón de procesos, por lo que el usuario en primera instancia no requerirá modelar sus procesos. Del mismo modo, se eliminó el componente WS-CDL ya que su función es coordinar la comunicación de negocio a negocio, una función que no es vital para una MiPyME.

C. Modelo de requisitos

Continuando con la metodología, se elaboraron los diagrama de casos de uso y diseño arquitectónico. Con el análisis del modelo de procesos y las tecnologías disponibles, se detectó un conjunto de requisitos necesarios para el ambiente que se busca diseñar. Los requerimientos identificados se muestran en la Figura 5.

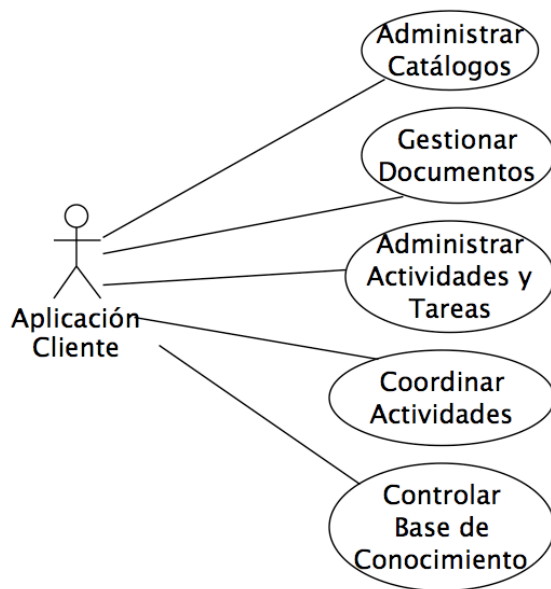


Figura 5. Casos de uso de fTIMoN.

Tales casos de uso representan la funcionalidad que busca cumplir la aplicación Flujos de Trabajo para la Implementar MoProSoft como Norma (fTIMoN). La definición general de cada uno de éstos se describe a continuación:

--**Administrar catálogos.** Este agrupa las altas, bajas, y modificaciones de catálogos de empleados, usuarios, proyectos, y clientes.

--**Gestionar documentos.** En este caso de uso se contempla el llenado, almacenamiento y consulta de los documentos que debe generar la organización en los procesos definidos por MoProSoft.

--**Administrar actividades y tareas.** Representa la habilidad de los distintos roles para consultar sus actividades correspondientes, y la alimentación de éstas en el desarrollo de los procesos.

--**Coordinar actividades.** Caso de uso que expresa el manejo de diagramas de actividades (Figura 2), es decir, la persistencia en la secuencia de actividades, sus dependencias, roles involucrados, así como productos de entrada y salida.

--**Controlar la base de conocimiento.** Incluye la función para aprovechar la base de conocimiento que se genera con las actividades definidas en MoProSoft, y las aportaciones de los empleados de la organización.

En la siguiente sección, se explicará a detalle la arquitectura de la aplicación fTIMoN como parte del modelo de requisitos.

IV. ARQUITECTURA DE FTIMON

Siendo MoProSoft un modelo de procesos, se consideró tomar el estándar de BPMI como base para la arquitectura de fTIMoN. La arquitectura propuesta se muestra en la Figura 6.

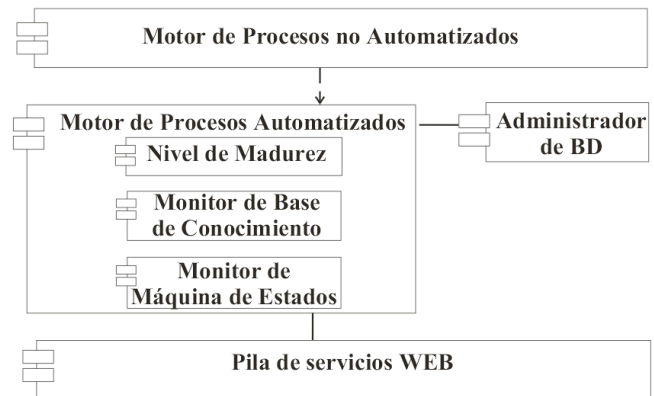


Figura 6. Arquitectura de fTIMoN.

Componentes de la arquitectura:

--**Motor de procesos no automatizados.** Como su nombre lo indica es un componente encargado de los procesos que requieren la interacción con los usuarios, identifica las personas y roles encargadas de realizar las actividades, y de la transición entre procesos automatizados y humanos.

--**Motor de procesos automatizados.** Componente central que coordina la comunicación asíncrona entre los servicios, esta designado para procesos de negocio automatizado, rutea mensajes, entre otras funciones.

--**Nivel de madurez.** Este componente se encargará del manejo de niveles de madurez definidos en MoProSoft 1.3.

--**Monitor de máquina de estados.** La importancia de este componente es por llevar el control de los distintos estados por los que puede pasar cada una de las actividades de los distintos procesos del modelo.

--**Monitor de base de conocimiento.** Componente encargado de obtener información perteneciente a la base de datos y que sea de utilidad para los roles y la actividad que realizan.

--**Administrador de base de datos.** Sirve de repositorio de los procesos, archivos de lenguaje de ejecución de procesos, base de conocimiento, mejores prácticas, y es quien ofrece la explotación de tal información.

--**Pila de servicios WEB.** La función de esta pila se centra en atender a las distintas aplicaciones cliente que desee consultar y/o manipular la información procesada dentro del servidor.

Como se aprecia en la Figura 6, la mayor aportación de fTIMoN son el coordinar la secuencia y dependencia de tareas y productos internos, de entrada y salida, dar a conocer entre los roles los estados de las actividades dependiendo del nivel de madurez de los procesos y la categoría de los mismos. Asimismo, la funcionalidad de una pila de servicios WEB para atender distintas aplicaciones cliente que deseen consultar y/o manipular información de la base de conocimiento. Es decir, fTIMoN se enfoca en el accionar de los diagramas de actividades y trabajo en grupo, puntos de los que carecen las aplicaciones actuales que dan soporte para la implementación de MoProSoft, especificadas en la sección II.

V. CONCLUSIONES

A partir de que la industria mexicana de software posee un modelo de procesos basado en las mejores prácticas reconocidas a nivel mundial y mecanismos de evaluación, se detectó la necesidad de contar con aplicaciones orientadas a la interpretación e implantación de dicho modelo. Sin embargo, hasta el momento existen las aplicaciones Kuali y MDM las cuales están orientadas sólo a la parte estática del patrón de procesos y en algunos casos no toman en cuenta todos los procesos definidos.

En este documento, se propuso una reestructuración de los diagramas de actividad tomando en cuenta el nivel de madurez de los procesos de la organización y la naturaleza de sus productos. También, se diseñó una representación que despliegue la carga de actividades para identificar el número de actividades por rol dentro de un proceso dependiendo el nivel de madurez. Se describió el diseño arquitectónico de fTIMoN el cual se centra en el patrón de procesos y los diagramas de actividades definidos en MoProSoft.

El propósito de este trabajo es presentar un diseño arquitectónico que sirva de base para el desarrollo de un sistema que cuente como núcleo un motor ejecute los procesos de MoProSoft.

Por último, se reconoce la activa participación de Gobierno del Estado de Baja California (a través de la Dirección General de Informática), los miembros del Cluster IT@Baja y los académicos de diversas instituciones en el interés para adoptar a MoProSoft como un modelo de referencia en sus procesos de software.

AGRADECIMIENTO

Los autores quisieran agradecer muy especialmente las aportaciones y sugerencias de parte de los profesores Gloria E. Chavez Valenzuela, Gabriel A. López Morteo, ambos de la Universidad Autónoma de Baja California y Hanna Oktaba, de la Facultad de Ciencias de la Universidad Autónoma de México, para el desarrollo de este trabajo.

REFERENCIAS

- [1] R. A. Vazquez. PYMES y la vanguardia tecnológica en sistemas de información. 2002.
- [2] PROSOFT, Programa para el Desarrollo de la Industria de Software. 3er. Trimestre 2005. México.

- [3] A. Hualde y R. Gomis. 2004. La construcción de un Cluster de software en la frontera noroeste de México. *Revista Frontera Norte*. 32. Pág. 7-34. Julio 2004. México.
- [4] D. A. Caballero De La Villa. *Manejador de Documentos MoProSoft*. Tesis de Licenciatura. Ingeniería en Sistemas Computacionales. Universidad de las Américas, Puebla. 2005.
- [5] H. Oktaba, C. Alquicira Esquivel, A. Su Ramos, A. Martínez Martínez, G. Quintanilla Osorio, M. Ruvalcaba López, F. López Lira Hinojo, M. E. Rivera López, M. J. Orozco Mendoza, Y. Fernández Ordóñez, M. A. Flores Lemus. *Modelo de Procesos para la Industria de Software*. Versión 1.1. 2003. México.
- [6] *Diario Oficial de la Federación*. Secretaría de Gobernación. Pág. 40. México. Disponible en: http://www.gobernacion.gob.mx/dof/2005/agosto/dof_15-08-2005.pdf Consultado en Febrero de 2006.
- [7] H. Oktaba, C. Alquicira Esquivel, A. Su Ramos, A. Martínez Martínez, G. Quintanilla Osorio, M. Ruvalcaba López, F. López Lira Hinojo, M. E. Rivera López, M. J. Orozco Mendoza, Y. Fernández Ordóñez, M. A. Flores Lemus. *Modelo de Procesos para la Industria de Software*, por niveles de capacidad de procesos. Versión 1.3. 2005. México.
- [8] Kuali. México. Disponible en: <http://foros.software.net.mx/kuali/> Consultado en Febrero de 2006.
- [9] H. Oktaba. *Tejiendo nuestra Red*. MoProSoft nuestra ventaja competitiva. *Revista Software Guru Conocimiento en Práctica*. Vol. 1. Num. 5. Pág. 6. 2005. México.
- [10] H. Osher, W. Harrison y P. Tarr. *Software Engineering Tools and Environments: a Roadmap*. International Conference on Software Engineering. Limerick, Irlanda. Pág. 261- 277. 2000.
- [11] J. García Molina, M. J. Ortín, B. Moros y J. Nicolás. *De los Procesos de negocio a los casos de uso*. *Jornadas de Ingeniería del Software y Bases de Datos*. Pág. 103-116. 2000.
- [12] H. E. Eriksson y M. Penker. *Business Modeling with UML*. Business Patterns at work. John Wiley & Sons, Inc. 2000.
- [13] M. Havey. *Essential Business Process Modeling*. Sebastopol, CA. O'Reilly Media, Inc. 2005.
- [14] WfMC. *Workflow Reference Model*. 1995. Disponible en: <http://www.wfmc.org> Consultado en Enero de 2006