

Actas de los Talleres de las Jornadas de Ingeniería del
Software y Bases de Datos, Vol. 4, No. 4, 2010

III Taller de Procesos de Negocio e Ingeniería de Servicios (PNIS 2010)

7 de Septiembre de 2010
Valencia, España

Organizadores del Taller:

Manuel Resinas (Universidad de Sevilla)
Antonio Ruiz-Cortés (Universidad de Sevilla)
Joan Antoni Pastor (Universitat Oberta de Catalunya)
María Ribera Sancho (Universitat Politècnica de Catalunya)

Comités del III Taller de Procesos de Negocio e Ingeniería de Servicios (PNIS 2010)

Comité Organizador

Manuel Resinas (Universidad de Sevilla)
Antonio Ruiz-Cortés (Universidad de Sevilla)
Joan Antoni Pastor (Universitat Oberta de Catalunya)
María Ribera Sancho (Universitat Politècnica de Catalunya)

Comité de Programa

Acuña, Silvia T. (Univ. Autónoma de Madrid)
Botella, Pere (Univ. Politècnica de Catalunya)
Canós, Jose Hilario (Univ. Politècnica de Valencia)
De Castro, Maria Valeria (Univ. Rey Juan Carlos)
Durán, Amador (Universidad de Sevilla)
Fabra Caro, Francisco Javier (Universidad de Zaragoza)
García, Félix (Univ. de Castilla-La Mancha)
Lopez Cobo, Jose M.
Murillo, Juan Manuel (Univ. Extremadura)
Pedreira, Óscar (Univ. Coruña)
Pelechano, Vicente (Univ. Politècnica de Valencia)
Penadés, Maria del Carmen (Univ. Politècnica de Valencia)
Peña, Joaquín (Univ. de Sevilla)
Ruiz, Francisco (Univ. de Castilla-La Mancha)

Tabla de contenidos

Sesión 1. Mejora y monitorización de procesos

Mejora continua de procesos de negocio basada en PmCompetisoft integrando BPMM	1
<i>Andrea Delgado, Francisco Ruiz, Ignacio García-Rodríguez de Guzmán</i>	
Challenges To Support A PPI Management Lifecycle	2
<i>Adela Del Río Ortega, Manuel Resinas, Antonio Ruiz Cortés</i>	
Usando técnicas BPM para agilizar la gestión de procesos software y mejorar la alineación con el negocio	3
<i>Jose Javier Berrocal Olmeda, Jose García-Alonso, Juan Manuel Murillo Rodriguez</i>	

Sesión 2. Gestión de procesos de negocio

BP Research Lines at PROS	4
<i>Victoria Torres, Pau Giner, Vicente Pelechano</i>	
Hints on how to face business process compliance	5
<i>Cristina Cabanillas, Manuel Resinas, Antonio Ruiz-Cortés</i>	
Aplicación del Paradigma de las Redes-en-Redes en los Entornos de Procesos de Negocio	6
<i>Javier Fabra, Joaquín Ezpeleta</i>	

Mejora continua de procesos de negocio basada en PmCompetisof integrando BPMM

Andrea Delgado

Instituto de Computación
Facultad de Ingeniería
Universidad de la República
1300 Montevideo, Uruguay
adelgado@fing.edu.uy

Francisco Ruiz

Grupo Alarcos
Departamento de Tecnologías y
Sistemas de Información
Universidad de Castilla – La Mancha
13071 Ciudad Real
francisco.ruizg@uclm.es

Ignacio García-Rodríguez
de Guzmán

Grupo Alarcos
Departamento de Tecnologías y
Sistemas de Información
Universidad de Castilla – La Mancha
13071 Ciudad Real
ignacio.grodriguez@uclm.es

Resumen

Durante muchos años la mejora de los procesos de desarrollo y mantenimiento de software se ha estudiado desde un punto de vista de ingeniería, con la premisa de que un mejor proceso de desarrollo permite la construcción de mejores productos. En este contexto son bien conocidos modelos de madurez como CMM, CMMI, ISO/IEC 15504 o ISO 9001:2000. Más recientemente ha sido propuesto COMPETISOFT, especialmente concebido para pequeñas y medianas empresas. Por otro lado, los procesos de negocio representan la forma en que las organizaciones funcionan para alcanzar sus objetivos de negocio, especificando la secuencia de actividades a realizar, y quién y en qué momento las realiza a fin de proporcionar resultados de valor a sus clientes. Desde un punto de vista diferente al de ingeniería de software, un proceso de desarrollo de software puede ser visto también como un proceso de negocio que realiza una organización entre cuyos objetivos está desarrollar y mantener un producto software. Por este y otros motivos, OMG se ha basado en CMM y CMMI para elaborar el “Business Process Maturity Model” (BPMM), que pretende proporcionar pautas de mejora para el ámbito de los procesos de negocio, semejantes a las propuestas en el ámbito de la ingeniería de software. En este artículo se presenta una adaptación a procesos de negocio del proceso de mejora PmCOMPETISOFT integrando BPMM.

1. Motivación

El objetivo de la Gestión de Procesos de Negocio (Business Process Management, BPM) [1][2] es ayudar a una organización en la gestión de sus

procesos de negocio en base a su ciclo de vida [3], desde su definición, modelado, validación, simulación, ejecución, medición hasta su evaluación. Esta última fase cobra importancia en el contexto de los esfuerzos de mejora que las organizaciones emprenden para mejorar sus procesos de negocio, analizando los datos obtenidos de la ejecución de dichos procesos para identificar oportunidades de mejora utilizando técnicas como la minería de procesos (Process Mining) [4], y herramientas como ProM [5]. Pero, como sucede también en el ámbito de desarrollo de software, cuando no se sigue un proceso de mejora, los esfuerzos de mejora pueden o no tener éxito, dependiendo sobre todo de las personas que los realizan.

Nuestro objetivo general es apoyar la mejora continua de procesos de negocio mediante la definición del marco MINERVA (Model driven & sService oriented framework for the continuous business process improvement & related tools) [6]. Este framework integra los paradigmas de Computación Orientada a Servicios (Service Oriented Computing, SOC) [7] y el Desarrollo Dirigido por Modelos (Model Driven Development, MDD) [8] para apoyar la generación automática de servicios desde procesos de negocio, separando la definición de los procesos de negocio de su implementación técnica. MINERVA se compone de tres dimensiones: conceptual [9], metodológica [10] y de herramientas [11], incluyendo conceptos e ideas, metodologías y herramientas de soporte respectivamente. La dimensión metodológica incluye el proceso de mejora continua de procesos de negocio (Business Process Continuous Improvement Process, BPCIP). BPCIP está basado en PmCOMPETISOFT [12], que es la propuesta de mejora de procesos para guiar los esfuerzos de mejora en las organizaciones de

desarrollo de software, definido en el proyecto COMPETISOFT [13]. Para darle el barniz adecuado al ámbito de los procesos de negocio, hemos adaptado PmCOMPETISOFT integrando el uso del reciente estándar BPMM [14] de OMG, que se basa en el Capability Maturity Model (CMM) [15] y en el Capability Maturity Model Integration (CMMI) [16], para proporcionar las mismas pautas para la mejora y madurez de procesos de negocio.

El resto de este artículo se organiza de la siguiente manera: en las secciones 2 y 3 se resumen, respectivamente, el proceso de mejora PmCOMPETISOFT y el estándar BPMM. En la sección 4 se presenta el proceso de mejora continua de procesos de negocio (BPCIP) que proponemos y se finaliza con conclusiones y trabajo futuro en la sección 5.

2. Proceso de mejora PmCompetisoft

PmCOMPETISOFT [12] es un proceso para guiar la mejora de procesos software en pequeñas empresas. Ha sido desarrollado como parte de COMPETISOFT[13], un proyecto iberoamericano que incluye un modelo de referencia de procesos, un modelo de evaluación y un modelo de mejora. Dichos modelos fueron desarrollados teniendo en cuenta las normas existentes y referencias tales como CMMI, ISO/ IEC 15504 o ISO 9001:2000. PmCOMPETISOFT integra el modelo de mejora, y es un proceso explícito que proporciona una guía, paso a paso, para llevar a cabo esfuerzos de mejora de procesos. La Figura 1 muestra una

visión global del marco metodológico general de COMPETISOFT tomado de [12], con foco en el modelo de mejora en que está incluido PmCOMPETISOFT.

PmCOMPETISOFT define las actividades, roles y productos de trabajo para guiar el esfuerzo de mejora en la organización que hace software. Los principios rectores de PmCOMPETISOFT son los siguientes [12]: "(i) obtención temprana y continua de mejoras, (ii) diagnóstico rápido y continuo del proceso, (iii) medición elemental del proceso, (iv) colaboración y comunicación en grupo efectiva, y (v) aprendizaje continuo". La norma ISO/IEC 15504-4, y los modelos IDEAL y SCRUM fueron tenidos en cuenta al definir PmCOMPETISOFT, y varias de sus prácticas de mejora fueron adaptadas e integradas.

Define una primer actividad de "Puesta en marcha del ciclo", que tiene como objetivo crear una propuesta de mejora, que esté alineada con el plan estratégico de la organización, y cuyo fin es guiar a la organización a través de las actividades del ciclo. En segundo lugar la actividad de "Diagnóstico de procesos" en la que se realiza una evaluación de procesos para descubrir el estado general de los procesos de la organización, definiendo una priorización entre ellos. La priorización se usa para planificar en que iteración se llevará a cabo cada mejora. Los modelos de referencia de procesos y de evaluación de COMPETISOFT se utilizan para el diagnóstico de procesos, pero cualquier otro modelo también podría ser utilizado, como CMMI.

En la actividad de "Formulación de mejoras" la iteración actual del plan de mejora se diseña y

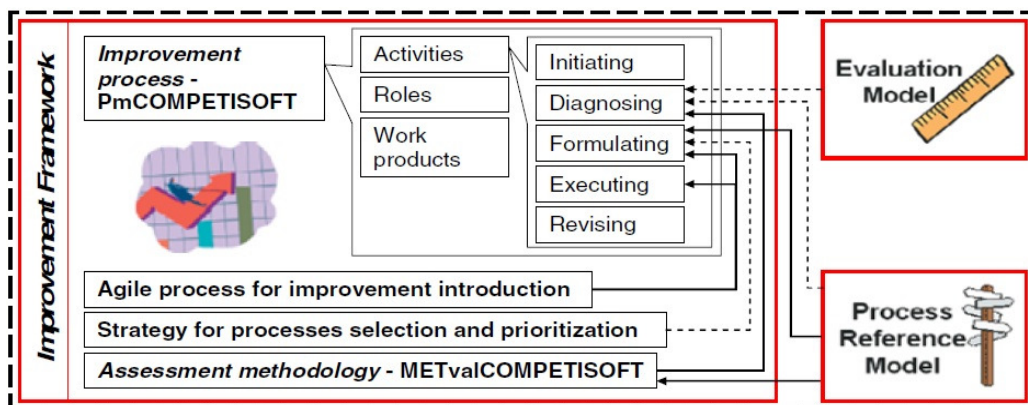


Figura 1. Marco metodológico general de COMPETISOFT de [12]

planifica, definiendo la forma de incorporar las mejoras en el proceso. "Ejecución de mejoras" es la actividad en la que las mejoras para la iteración actual son gestionadas y ejecutadas, siguiendo los planes establecidos. Las mejoras introducidas en los procesos seleccionados se analizan, y los resultados, el desempeño y la evaluación de la iteración se registran. Por último, la actividad de "Revisión del ciclo" tiene como objetivo realizar un análisis post-mortem del ciclo de mejora, registrando datos tales como los procesos que se han mejorado en el ciclo, y cualquier otra información pertinente.

3. Modelo de Madurez de Procesos de Negocio (BPMM)

El estándar BPMM [14] de OMG está basado en los principios y prácticas de los modelos CMM y CMMI para mejora de procesos de desarrollo de software, y ha sido desarrollado por co-autores de dichos modelos. Su objetivo es proporcionar un marco de referencia para organizar los pasos a seguir para la mejora continua de los procesos en cinco niveles de madurez, que establecen las bases para el esfuerzo de mejora. En cada nivel de madurez se implementan prácticas clave, con lo que el progreso entre los niveles es posible tomando pequeños pasos desde los niveles inferiores a los niveles superiores. Siguiendo los modelos CMM y CMMI, se definen cinco niveles de madurez, que se muestran en la Tabla 1.

En cada nivel de madurez se definen un conjunto de áreas de proceso (Process Areas), cada una de las cuales contiene prácticas que cuando implementan juntas proporcionan una capacidad de proceso que contribuye al nivel de madurez. BPMM define treinta áreas de proceso en total: nueve para el nivel dos, diez para el nivel tres, cinco para el nivel cuatro y seis para el nivel cinco. Cada una se compone de varios elementos como el propósito del área, objetivos específicos y de institucionalización, prácticas específicas y de institucionalización que pueden tener sub-prácticas, y guías para las prácticas. Algunas de estas guías se han definido para ayudar en las actividades de medición incluidas en las áreas de procesos para definir, especificar, recoger, almacenar y verificar las medidas para el análisis del esfuerzo de mejora y validación.

Tabla 1. Niveles de madurez de BPMM [14]

Nivel de Madurez	Foco	Salida
5 Innovativo	Implementar mejora proactiva continua para alcanzar objetivos del negocio	Innovación planificada, gestión de cambios, procesos capaces
4 Predecible	Gestionar procesos y resultados cuantitativamente y explotar beneficios de la estandarización	Procesos estables, gestión del conocimiento y reuso, resultados predecibles
3 Estandarizado	Definición de medidas estándar de procesos, entrenamiento en ofertas de servicios y productos	Crecimiento de la productividad, automatización efectiva, economía de escala
2 Gestionado	Gestión disciplinada de unidades de trabajo para estabilizar el trabajo y controlar los compromisos	Prácticas repetibles, reducción del re-trabajo, compromisos satisfechos
1 Inicial	Motivar a las personas para superar los problemas y "realizar el trabajo"	Crecimiento de la productividad, automatización efectiva, economía de escala

En BPMM; al igual que en CMM y CMMI, la madurez de un proceso es una medida del grado en que los procesos están definidos explícitamente, gestionados, medidos, controlados y son eficaces. La capacidad de un proceso refiere al rango de los resultados esperados que se pueden obtener siguiendo dicho proceso, proporcionando la base para predecir sus resultados más probables. Los niveles de madurez y las áreas de proceso (Process Areas) definidos en BPMM son indicadores de la capacidad de los procesos. La madurez de un proceso implica que la capacidad del proceso ha mejorado en el tiempo. Por otro lado, el desempeño de un proceso describe los resultados reales que se obtienen al ejecutar un proceso. A medida que el proceso consigue la realización de diferentes áreas de proceso que corresponden a cada nivel de madurez, éste madura con el tiempo, dando lugar a una organización más madura que puede manejar sus procesos y predecir sus resultados futuros, mejorándolos continuamente en base a medidas de datos consistentes, su recolección y análisis.

4. Proceso de Mejora Continua de Procesos de Negocio (BPCIP)

En este apartado se presenta el Proceso de Mejora Continua de Procesos de Negocio (BPCIP) que proponemos. Es parte de la dimensión metodológica de MINERVA, y su principal meta es guiar la mejora continua de los procesos de negocio integrando el estándar BPMM y su ciclo de vida[3]. En la misma dimensión de MINERVA también se incluye la Metodología de Procesos de Negocio Orientada a Servicios (Business Process Service Oriented Methodology, BPSOM) [10] para guiar el desarrollo orientado a servicios desde los procesos de negocio, que se utiliza en el ciclo de vida de los procesos de negocio para navegar de la fase de Diseño y Análisis a la de Configuración. La propuesta general se ilustra en la Figura 2, donde se muestra como se combina la ejecución del ciclo de vida de los procesos de negocio con el ciclo de mejora de PmCOMPETISOFT que integra al BPMM. Esta combinación da como resultado un “macro” ciclo de mejora continua de los procesos de negocio que permite incorporar las mejoras detectadas en forma sistemática y con resultados predecibles.

Antes de introducir un esfuerzo de mejora sobre un proceso de negocio es necesario contar con medidas y datos de su modelo y ejecución, así como su capacidad según BPMM. Estas medidas deben definirse y recogerse desde las primeras etapas de su ciclo de vida: medidas sobre el diseño (modelos) y ejecución de los procesos de negocio [17] [18] e Indicadores Clave del Desempeño (Key Performance Indicators, KPIs) deben especificarse en la definición de los procesos de negocio a fin de determinar con claridad qué medidas es necesario recoger y cómo.

4.1. Ejecución del ciclo de vida de los procesos de negocio

En la primera fase de Diseño y Análisis del ciclo de vida de los procesos de negocio, el objetivo es modelar los procesos de negocio de la organización, integrando medidas de diseño para validar los modelos. En la segunda fase de Configuración, los procesos de negocio modelados son implementados y desplegados en la organización en las plataformas seleccionadas. Para navegar desde la fase de Diseño y Análisis a la de Configuración, la metodología BPSOM guía

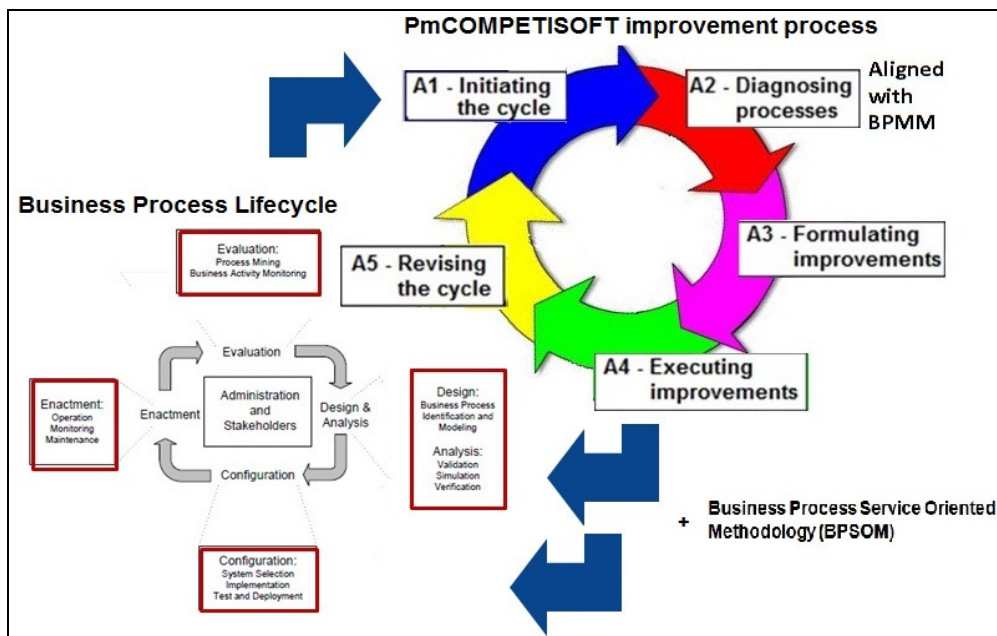


Figura 2. Propuesta general de mejora de procesos de negocio de MINERVA

el desarrollo orientado a servicios desde los procesos de negocio, incluyendo transformaciones QVT [19] para la generación automática de modelos de servicios en SoaML [20] desde modelos de procesos de negocio en BPMN [21]. En la tercera fase de Ejecución los procesos de negocio se ejecutan y los datos definidos para evaluar esta ejecución se almacenan en los archivos de log de ejecución. En la última fase del ciclo de vida de procesos de negocio, la fase de Evaluación, los datos recogidos en la etapa anterior de Ejecución son analizados mediante técnicas como minería de procesos para identificar oportunidades de mejora. En este punto, cuando se detecta una oportunidad de mejora, se inicia el ciclo de mejora de procesos de negocio guiado por PmCOMPETISOFT para llevar a cabo el esfuerzo de mejora, que se presenta a continuación.

4.2. Ejecución de PmCOMPETISOFT

Una vez que se analizaron los datos de los logs de ejecución de los procesos de negocio en la fase de Evaluación de su ciclo de vida, es posible detectar oportunidades de mejora en uno o varios procesos de negocio de la organización. En este punto, se decide realizar un esfuerzo de mejora sobre determinados procesos de negocio, conociendo por lo tanto su desempeño en base a dichos datos. La propuesta de mejora que será llevada adelante, se define en la ya mencionada primer actividad de PmCOMPETISOFT "A1-Poner en marcha el ciclo". En la actividad "A2-Diagnóstico de procesos", como se presentó anteriormente, se debe determinar la capacidad de los procesos bajo mejora, para lo cual cambiamos la utilización que hace PmCOMPETISOFT de los modelos de referencia y evaluación de procesos de COMPETISOFT, por el uso del estándar BPMN.

Al utilizar el modelo de evaluación para evaluar el proceso, y en base al modelo de referencia de procesos de BPMN, es posible descubrir la medida en que el proceso cumple cada área de proceso (Process Area), definiendo la capacidad del proceso y el nivel de madurez en que se encuentra. Al contar con datos del desempeño de los procesos de negocio, se puede suponer que el punto de partida sea ya el nivel 2, donde se tiene información (aunque sea mínima) de la ejecución de los procesos. A medida que se va subiendo de nivel la calidad de las medidas mejora, definiendo por ejemplo Indicadores Clave de Desempeño (Key Performance Indicators,

KPIs) a partir del nivel 3 de forma de obtener mayor información sobre la ejecución de los procesos. Esto permite a partir del nivel 4, poder predecir la ejecución de los procesos de negocio en forma estadística según la varianza de las distintas ejecuciones, y en el nivel 5 se cuenta ya con datos que permiten innovar en los procesos de negocio donde sea necesario.

Por lo tanto, al realizar el diagnóstico de los procesos bajo mejora, es posible encontrar incluso más oportunidades de mejora según el cumplimiento de las áreas de procesos, las que habrá que añadir al esfuerzo de mejora que se llevará adelante. En la actividad "A3-Formulación de mejoras" se define cómo se va a realizar la mejora en dichos procesos, esto es, sabiendo lo que se quiere mejorar en los procesos para progresar al nivel siguiente de BPMN, establecer cómo (haciendo qué) se mejorará la actividad, parte del flujo, manejo de datos, implementación asociada, etc. que se quiere mejorar. Las oportunidades de mejora que corresponden a la iteración actual son gestionadas y ejecutadas en la actividad "A4-Ejecutar mejoras". Para poder ejecutar estas mejoras, se reingresa nuevamente al ciclo de vida de los procesos de negocio en la fase apropiada, para introducir las mejoras en los procesos y poder ejecutar, medir y evaluar la nueva versión del proceso de negocio generada con las mejoras incorporadas.

4.3. Reingreso a la ejecución del ciclo de vida de los procesos de negocio

Para introducir las mejoras detectadas de la forma definida en los procesos de negocio, se debe reingresar al ciclo de vida de los procesos de negocio ejecutándolo nuevamente desde la fase apropiada. Si la mejora requiere cambios que afectan al modelo de proceso de negocio, se reingresa al ciclo de vida en la fase de Diseño y Análisis, donde el modelo existente será modificado según las mejoras que se hayan detectado. Sin embargo, si la mejora se refiere a aspectos relacionados con la ejecución, tales como cambios en la implementación de los servicios que realizan los procesos de negocio, se reingresa al ciclo en la fase de Configuración. En cualquier caso, se genera una nueva versión del proceso de negocio que se despliega y pone en ejecución en la organización, obteniendo así nuevos datos de la ejecución del proceso, para ser analizados en la fase de Evaluación, que se muestra en la Figura 3.

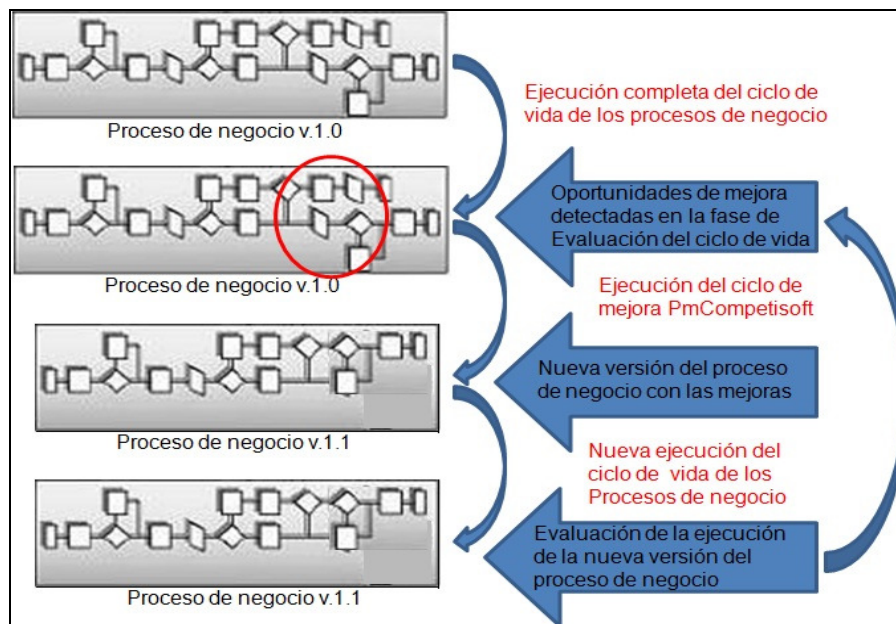


Figura 3. Ejecución de ciclos de vida y mejora de procesos de negocio en MINERVA

Los datos obtenidos de la ejecución de la nueva versión del proceso de negocio con las mejoras introducidas, serán comparados con los datos anteriores que originaron la detección de las oportunidades de mejora, con el fin de analizar si la mejora se ha introducido de forma satisfactoria en el proceso. Si es así, el nuevo proceso es aceptado y establecido en la organización, si no, nuevas oportunidades de mejora se podrán detectar y se llevará a cabo un nuevo ciclo de mejora, repitiendo nuevamente la ejecución de PmCompetisoft descrita. El análisis comparativo y los resultados, junto con la realización y evaluación de la iteración actual de las mejoras implementadas serán registradas. Finalmente, en "A5- Revisión del ciclo" se evalúa el desarrollo del propio proceso de mejora.

5. Conclusiones y trabajo futuro

El Proceso de Mejora Continua de Procesos de Negocio (BPCIP) se define en el contexto del marco MINERVA y tiene como objetivo guiar los esfuerzos de mejora de procesos de negocio en organizaciones de forma sistemática, para definir, ejecutar y evaluar las oportunidades de mejora detectadas. Se basa en una adaptación del proceso de mejora PmCOMPETISOFT integrando la

norma BPMM de OMG para determinar la capacidad y la madurez de los procesos de negocio, y los pasos a seguir para su mejora.

Se ha presentado el macro ciclo de mejora continua que define BPCIP, mostrando la relación entre la utilización del proceso de mejora PmCOMPETISOFT para la introducción de las mejoras, y el ciclo de vida de los procesos de negocio que debe ser ejecutado cada vez que se introduce una mejora en un proceso y se obtiene una nueva versión del mismo. Esta nueva versión será ejecutada, medida y evaluada de forma de poder comparar con las medidas anteriores y así determinar si la mejora fue introducida con éxito.

Adicionalmente, la implementación de procesos de negocios en el marco MINERVA se realiza con servicios, siguiendo la metodología BPSOM, incluyendo la generación automática de modelos de servicio a partir de modelos de procesos de negocio, separando la definición de los procesos de negocio de su implementación técnica. Esto permite que las mejoras puedan ser introducidas en cada área (negocio, tecnología) con mínimo impacto en la otra, aportando al soporte del ciclo de mejora e incorporación ágil de cambios tanto en la definición de los procesos de negocio como en su implementación en diferentes tecnologías.

Creemos que BPCIP puede ser una guía útil para la mejora continua de los procesos de negocio en las organizaciones, basada en la introducción sistemática de mejoras a los procesos de negocio en base a las definiciones presentadas en la propuesta. Nuestro trabajo futuro central está enfocado en detallar completamente BPCIP y realizar un caso de estudio para validar de manera experimental la propuesta.

Agradecimientos. Este trabajo ha sido parcialmente financiado por la Agencia Nacional de Investigación e Innovación (ANII) de Uruguay, proyecto ALTAMIRA (Junta de Comunidades de Castilla-La Mancha, España, Fondo Social Europeo PII2I09-0106-2463) y proyecto PEGASO/MAGO (Ministerio de Ciencia e Innovación MICINN, España, y Fondo Europeo de Desarrollo Regional FEDER, TIN2009-13718-C02-01).

Referencias

- [1] Smith, H., Fingar, P., Business Process Management: The third wave, Meghan-Kieffer, (2003)
- [2] van der Aalst, W.M.P., ter Hofstede, A., Weske, M., Business Process Management: A Survey, In: International 3 Conference on Business Process Management, (2003)
- [3] Weske, M., BPM Concepts, Languages, Architectures, Springer, (2007)
- [4] van der Aalst, W.M.P., Reijers, H. A., Medeiros, A., Business Process Mining: an Industrial Application, Information Systems Vol.32 Issue 5, 713-732, (2007)
- [5] ProM, Process Mining Group, Eindhoven University of Technology, The Netherlands, <http://prom.win.tue.nl/research/wiki>
- [6] Delgado A., Ruiz F., García-Rodríguez de Guzmán I., Piattini M., MINERVA: Model driven and service oriented framework for the continuous business processes improvement & related tools, In: 5th IW on Engineering Service-Oriented Applications (WESOA'09), Stockholm, November (2009).
- [7] Papazoglou, M.; Traverso, P.; Dustdar, S.; Leymann, F.: Service-Oriented Computing: State of the Art and Research Challenge, IEEE Computer Society, (2007)
- [8] Mellor, S., Clark, A., Futagami, T., Model Driven Development - Guest editors introduction, IEEE Computer Society, September/October (2003).
- [9] Delgado, A., Ruiz, F., García - Rodríguez de Guzmán, I., Piattini, M.: Towards an ontology for service oriented modeling supporting business processes, 4th Int. Conf. on Research Challenges in Information Science (RCIS'10), Niza, (2010).
- [10] 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, 2nd Int. Conf. on Business Process and Services Computing (BPSC'09), Leipzig, (2009).
- [11] 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)
- [12] Pino, F., Hurtado, J., Vidal, J., García, F., Piattini, M.: A Process for Driving Process Improvement in VSEs, Int. Conf. on Software Process (ICSP'09), Vancouver, (2009)
- [13] Oktaba, H., Garcia, F., Piattini, M., Pino, F., Alquicira, C., Ruiz, F.: Software Process Improvement: The COMPETISOFT Project. IEEE Computer 40(10), 21-28 (2007)
- [14] Business Process Maturity Model (BPMM), OMG, <http://www.omg.org/spec/BPMM>
- [15] Paulk, M., Curtis, B., Chrissis, M., & Weber, C. (1993). Capability maturity model (CMM) for software, v.1.1. Software Engineering Institute (SEI), CMU/SEI-93-TR-024.
- [16] Capability maturity model (CMMI) for software engineering, v.1.1, SEI, (2002), CMM Integration (CMMI) for development (CMMI-DEV) v.1.2, SEI, (2006)
- [17] Mendling, J., Metrics for process models, Springer, 978-3-540-89223-6, (2008)
- [18] Sánchez, L., Delgado, A., Ruiz, F., García, F., Piattini, M.: Measurement and Maturity of Business Processes. Eds.: Cardoso, J., van der Aalst, W., Handbook of Research on Business Process Modeling, Information Science Reference (IGI Global), pp.532-556, (2009)
- [19] Query/Views/Transformations (QVT) v.1.0, OMG <http://www.omg.org/spec/QVT/1.0>, (2008)
- [20] Soa Modeling Language (SoaML), OMG, <http://www.omg.org/spec/SoaML/>
- [21] Business Process Modeling Notation (BPMN), OMG, <http://www.omg.org/spec/BPMN/>

Challenges To Support A PPI Management Lifecycle*

Adela del-Río-Ortega Manuel Resinas Antonio Ruiz-Cortés
ETS Ingeniería Informática ETS Ingeniería Informática ETS Ingeniería Informática
Univ. de Sevilla Univ. de Sevilla Univ. de Sevilla
adeladelrio@us.es resinas@us.es aruiz@us.es

Abstract

An important aspect in the business process lifecycle is the evaluation of business processes performance, since it helps organisations to define and measure progress towards their goals. Performance requirements on business processes can be specified by means of Process Performance Indicators (PPIs), as suggested in many methodologies and frameworks like, for instance, COBIT, ITIL or EFQM. As a consequence, it is convenient to integrate the management of PPIs into the whole business process lifecycle from its design to its evaluation, enabling thus a more effective and efficient automated support to extract information from such indicators. In this paper, we analyse some approaches related to this issue and identify the challenges that need to be faced.

1 Introduction

It is increasingly important to evaluate the performance of business processes. A key instrument to carry out this evaluation is by means of Process Performance Indicators (PPIs). A PPI is a measure that reflects the critical success factors of a business process defined within an organisation, in which its target value must be reached in a certain period and reflects the objectives pursued by the organisation with that business process. Note that we use PPI as a kind of Key Performance

*This work has been partially supported by the European Commission (FEDER), and SETI (TIN2009-07366); and project P07-TIC-2533 funded by the Andalusian local Government.

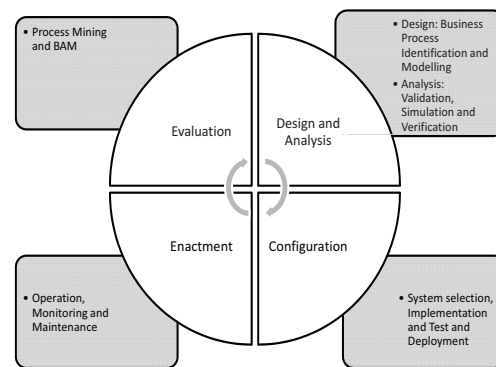


Figure 1: Business Process Lifecycle as described in [4]

Indicator (KPI) that focuses exclusively on the indicators defined on the business processes. Nowadays, many methodologies and frameworks like, for instance, COBIT, ITIL or the EFQM excellence model [1, 2, 3], confirm this importance by including the definition of these PPIs within their recommendations as a means to evaluate the performance of the existing business processes. In order to make this evaluation of business processes easier, it is convenient to integrate the management of PPIs into the whole business process lifecycle. In [4], the four-phase business process lifecycle of Figure 1 is proposed. This lifecycle starts with the Design and analysis phase, in which business processes are identified, modelled using a particular notation, validated to check whether all valid process instances are reflected by its corresponding business process model, simulated to detect possible undesired

execution sequences, and verified. Once the business process model is designed and verified, it needs to be implemented. This is done during the configuration phase, where, in case a dedicated software system is needed, it must be selected and configured in order to take into account the interactions of the employees with the system and the integration with existing software systems. Then, the implementation of business process needs to be tested and finally deployed in its target environment. Next phase is the enactment, that encompasses the run time of the business process. On the one hand, a correct orchestration is necessary for the business activities to be performed according to the business process's execution constraints. On the other hand, process monitoring is a good mechanism for providing information about the status of business process instances. Finally, the evaluation phase uses information available to evaluate and improve business process models and their implementations. Note that there not exists a strict temporal ordering in which these phases need to be executed; incremental and evolutionary approaches involving concurrent activities in multiple phases are, thus, common. In this paper, we are interested in describing what has been done in the literature regarding the integration of PPIs as first-class citizens in this business process lifecycle, and in outlining the open challenges this issue presents.

The remainder of this paper is organised as follows. In Section 2 we explain how to integrate PPIs into the business process management lifecycle and analyse how different approaches deal with this issue. From this section, a number of open research questions are identified and presented in Section 3. Finally, Section 4 draws the conclusions from our work, summarizes the paper and outlines our future work.

2 Integrating PPIs Into The Business Process Management Lifecycle

In order to make the evaluation of the performance of business processes easier, it is con-

venient to integrate the management of PPIs into the whole business process lifecycle. In this section, we take the business process lifecycle presented by Weske in [4, 5] as a reference and we describe how management of PPIs can be integrated into this lifecycle. Furthermore, we also report on several proposals that can be used to support this integration.

2.1 Design and Analysis

In the design and analysis phase, PPIs should be modelled together with the business process. Furthermore, this model of PPIs should also enable their analysis by detecting the dependencies amongst them at design time and also using them as part of the business process analysis, for instance in business process simulation techniques.

With respect to this phase, several authors have presented some approaches: Pedrinaci et al. present in [9] a metric ontology to allow the definition and computation of metrics. Popova et al. [13] present a framework for modeling performance indicators within a general organisation modeling framework. They define indicators by assigning values to a set of attributes. In [23], they point out the way these indicators are calculated and define relations between PPIs and the processes. Relationships between PPIs (causality, correlation and aggregation) are briefly introduced in [13] and explained in detail in [23]. They also define temporal properties over PPIs (called PI expressions) in [24]. Wetzstein et al. describe in [20] a KPI ontology using WSML to specify KPIs over semantic business processes. In [15] Mayerl et al. discuss how to derive metric dependency definitions from functional dependencies by applying dependency patterns. Castellanos et al. present in [18] the IBOM platform, that allows, among other things, to define business measures. The user can define business measures (through a GUI) to measure characteristics of process instances, processes, resources or of the overall business operations. Specifically, they characterize metrics through four attributes: name (unique), target entity (object to be measured), data type (numeric, boolean, taxonomy or SLA) and desirable met-

ric values. This approach is not focused on business processes but on the whole organisation.

2.2 Configuration

During the configuration phase, the instrumentation of the processes that are necessary to take the measures must be defined.

Momm et al. present in [19] a meta-model for the specification of the PPI monitoring along with an extension of the BPMN meta-model for the monitoring and an outline of methodology for an automated generation of this instrumentation. This approach explains how to define a KPI and how to take the measures.

2.3 Enactment

During the business process enactment, when valuable execution data is gathered, the PPIs' values have to be calculated and the monitoring of these PPIs should be carried out. For instance, this can be done based on execution logs that store information about the process such as the start or the end of activities.

To overcome this issue, Pedrinaci et al. present in [9] a Semantic Business Process Monitoring Tool called SENTINEL. This tool can support automated reasoning. Popova et al. propose in [23] formal techniques for analysis of executions of organizational scenarios. Wetzstein et al. introduce in [20] a framework for BAM as part of the semantic business process management, where PPI models are automatically converted to IT-level event-based models and used for real-time monitoring using reasoning technology. In this framework, there is a monitor model that specifies how events are processed to calculate KPIs, whose values are then published in dashboards

2.4 Evaluation

Finally, during the evaluation phase, the monitoring information obtained in the previous phase will help to identify correlations and predict future behaviour.

Castellanos et al. [18] address this issue, since their IBOM platform also allows to perform intelligent analysis on business measures to understand causes of undesired values and predict future values.

3 Challenges

As can be deduced from Section 2, business process lifecycle can benefit from the definition of PPIs over processes and from a set of analysis operations of them, enabling a more effective and efficient automated support to extract information from such indicators.

In order to obtain such benefit, there exists a number of research questions that are not answered yet. In the following we state some of them:

- **Which information is necessary in order to define Process Performance Indicators over business processes? How do these PPIs relate to the business processes for which they were defined?**

An appropriate definition of PPIs is key to enable the automated support of the aforementioned PPIs lifecycle. Moreover, in order to be able to get information, not only from the indicators, but also from the business processes they were defined for, it is important to define an explicit connection between indicators and the process elements they are measuring.

There are several research proposals to define PPIs, as presented in Section 2, but none of them are well-suited because either there are commonly used PPIs that cannot be defined with them or they are not ready to enable a design-time analysis of PPIs. In Pedrinati et al.'s approach [9] it is not clear whether the defined metrics are explicitly connected with the elements of a business process. Popova et al. [13] do not consider derived measures when defining indicators. Wetzstein et al. [20] do not take into account indicators related to data. Mayerl et al. [15] do not delve into the definition of measures,

they only set the semantics of some elements to consider when defining measures based on concepts of the CIM metrics model [16] (e.g. UnitOfWork, which relates metrics with functional entities) and the QoS UML profile described in [17]. Momm et al.'s metamodel for the specification of PPIs [19] lacks some properties identified like, for instance, the analysis period of the PPI). Finally, Castellanos et al. [18], as far as we can deduce from the paper, do not take into account during the definition of metrics some aspects such as the analysis period, the unit of measure or the dimension to be measured.

- **Which analysis operations can be applied to PPIs in order to get relevant information about them? Which information can be gathered from these PPIs' definition?**

Making queries and extracting information from PPIs may be useful when defining dashboards to monitor processes. Furthermore, the analysis of these PPIs at design-time can help to assure the definition of achievable goals or even identify conflicting goals. E.g. It could be detected if two PPIs inversely dependent are being tried to be optimised. Another benefit resulting from this analysis is the possibility to find out how certain activity influences PPIs or vice versa.

With respect to this challenge, there exist some approaches [9, 23, 20, 18] that address the analysis of PPIs, but they do it in runtime, losing the aforementioned benefits of this design-time analysis.

- **Which are the issues to consider when developing a suitable tool to support this automatic analysis of process performance indicators?**

Explicit business process models expressed in a graphical notation (such as, for instance, BPMN) ease communication about these processes, so that stakeholders can communicate efficiently, and refine and improve them. In the same way,

it is convenient to develop a graphical notation to depict these PPIs together with the corresponding business process models. Thus, it is necessary to develop a tool to support such a graphical notation of PPIs, as well as to enable their automated analysis. Furthermore, it would be very useful to integrate such a tool with an editor of business process diagrams.

To the best of our knowledge, there not exists such a graphical notation nor the described tool.

4 Conclusions and Future Work

In this paper, we argue the importance of integrating the management of PPIs into the whole business process lifecycle. Specifically, in the design and analysis phase, PPIs must be modelled together with the business process. This model should enable (at design-time) an automated or semi-automated analysis to, for instance, detect the dependencies and potential conflicts amongst them or to use them together with other business process analysis techniques such as simulation techniques. We have analysed how different approaches address this issue and their shortcomings. We have also identified the challenges that need to be faced to this respect.

Our future work focuses on finding answers to the open research questions presented in Section 3. In order to do so we have already started by defining an ontology for the definition of PPIs (described in a paper submitted to the 18th International Conference on Cooperative Information Systems-CoopIS 2010), whose main benefits can be summarised as follows:

1. The relation between PPIs and the business process is explicitly established.
2. It supports the definition of a wide variety of PPIs, including those associated with data objects. It also supports the definition of an expressive analysis period of a PPI

3. Dependencies between measures can be automatically obtained from the ontology, which enables the analysis of PPIs at design time. Furthermore, since the ontology has been defined in OWL DL, automated reasoners can be used to make queries about the PPI model

Moreover, we are extending this ontological definition with a textual language as closer to the natural language as possible, in order to ease this definition.

With respect to the second point, we plan to define a mechanism to make queries on PPIs (PPI-Q) similar to the one presented in [21] to make queries on BPMN (BPMN-Q). We are also working on describing mechanisms to derive restrictions (taking advantage from Constraint Satisfaction Problems) between PPIs, in order to extract knowledge in an automatic way from the PPIs defined over a particular business process.

Finally, to overcome the third challenge, we are developing a graphical notation in order to depict these ontological concepts of PPIs over business processes and we are also integrating this notation into the web-based editor ORYX [22] as a result of a collaboration stay with the BPT group at the HPI Potsdam.

References

- [1] ISACA: COBIT 4.1 (2009)
- [2] Office of Government Commerce: Information technology infrastructure library (ITIL) v3. Collection of books (2007)
- [3] European Foundation for Quality Management: EFQM excellence model 2010
- [4] Weske, M.: Business Process Management: Concepts, Languages, Architectures. Springer (2007)
- [5] del Río-Ortega, A., Resinas, M., Ruiz-Cortés, A.: Towards modelling and tracing key performance indicators in business processes. In: II Taller sobre Procesos de Negocio e Ingeniería de Servicios (PNIS 2009). (2009)
- [6] OMG: Business process modeling notation (BPMN) v2.0 (2009)
- [7] W3C OWL Working Group: OWL 2 Web Ontology Language (2009)
- [8] Brockmans, S., Volz, R., Eberhart, A., Löffler, P.: Visual modeling of OWL DL ontologies using UML. In: ISWC 2004. Volume 3298., Springer (2004) 198–213
- [9] Pedrinaci, C., Lambert, D., Wetzstein, B., van Lessen, T., Cekov, L., Dimitrov, M.: Sentinel: a semantic business process monitoring tool. In: Proceedings of the First International Workshop on Ontology-supported Business Intelligence (OBI). (2008)
- [10] Abramowicz, W., Filipowska, A., Kaczmarek, M., Pedrinaci, C., Starzecka, M., A., W.: Organization structure description for the needs of semantic business process management. In: 3rd international Workshop on Semantic Business Process Management colocated with 5th European Semantic Web Conference. (2008)
- [11] Abramowicz, W., Filipowska, A., Kaczmarek, M., Kaczmarek, T.: Semantically enhanced business process modelling notation. In: Proceedings of SBPM 2007 Semantic Process and Product Lifecycle Management in conjunction with the 3rd European Semantic Web Conference (ESWC 2007). (2007)
- [12] Pedrinaci, C., Domingue, J., Alves de Medeiros, A.K.: A core ontology for business process analysis. In: 5th European Semantic Web Conference. (2008)
- [13] Popova, V., Sharpanskykh, A.: Modeling organizational performance indicators. Information Systems (2009)
- [14] Garcia, F., Bertoa, M.F., Calero, C., Vallecillo, A., Ruiz, F., piattini, M., Genero, M.: Towards a consistent terminology for software measurement In: Information & Software Technology (48). 2006 631–644

- [15] Mayerl, C., Hüner, K., Gaspar, J.U., Momm, C., Abeck, S.: Definition of metric dependencies for monitoring the impact of quality of services on quality of processes. In: Second IEEE/IFIP International Workshop on Business-driven IT Management (Munich). (2007) 1–10
- [16] Distributed Management Task Force: Common information model (CIM)(2003)
- [17] OMG: UML TM profile for modeling quality of service and fault tolerance characteristics and mechanisms specification (2006)
- [18] Castellanos, M., Casati, F., Shan, M.C., Dayal, U.: IBOM: A platform for intelligent business operation management. In: Proceedings. 21st International Conference on Data Engineering, 2005., Hewlett-Packard Laboratories (2005) 1084– 1095
- [19] Momm, C., Malec, R., Abeck, S.: Towards a model-driven development of monitored processes. In: Wirtschaftsinformatik (2). (2007) 319–336
- [20] Wetzstein, B., Ma, Z., Leymann, F.: Towards Measuring Key Performance Indicators of Semantic Business Processes. In: 11th International Conference on Business Information Systems (BIS'08). Innsbruck, Austria (2008) 227–238
- [21] Awad, A., Decker, G., Weske, M.: Efficient compliance checking using bpmn-q and temporal logic. In: BPM. Volume 5240 of LNCS, Springer (2008) 326–341
- [22] Decker, G., Overdick, H., Weske, M.: Oryx - An Open Modeling Platform for the BPM Community. In: 6th International Conference on Business Process Management (BPM). Volume 5240 of LNCS. Springer (2008) 382–385
- [23] Popova V. and Sharpanskykh, A. Formal analysis of executions of organizational scenarios based on process-oriented specifications In: Applied Intelligence. (2009)
- [24] Popova V. and Sharpanskykh, A. Formal Goal-based Modeling of Organizationss In: MSVVEIS. (2008) 19–28

Usando técnicas BPM para agilizar la gestión de procesos software y mejorar la alineación con el negocio

Javier Berrocal

Dept. de Ingeniería de Sistemas
Informáticos y Telemáticos
Universidad de Extremadura
Avda. Universidad s/n
10071 Cáceres
jberolm@unex.es

José García-Alonso

Dept. de Ingeniería de Sistemas
Informáticos y Telemáticos
Universidad de Extremadura
Avda. Universidad s/n
10071 Cáceres
jgaralo@unex.es

Juan Manuel Murillo

Dept. de Ingeniería de Sistemas
Informáticos y Telemáticos
Universidad de Extremadura
Avda. Universidad s/n
10071 Cáceres
juanmamu@unex.es

Resumen

Actualmente, las compañías software utilizan los procesos de negocio con diversos propósitos. Por un lado, deben definir y gestionar sus procesos software, ya que son sus procesos de negocio. Sin embargo, la gestión de estos procesos conlleva la realización de una gran cantidad de actividades. Lo que requiere de un gran esfuerzo, e implica una disminución de la productividad de los gestores. El primer objetivo de nuestra investigación es reducir este esfuerzo. Para ello, se está adaptando un BPMS para que sea capaz de ejecutar procesos software y, así, automatizar el mayor número posible de estas actividades. Disminuyendo el esfuerzo y aumentando la productividad de los gestores. Por otro lado, algunas de estas compañías utilizan los procesos de negocio del cliente como artefacto inicial para desarrollar sus productos, ya que estos productos deben estar alineados con dichos procesos. Sin embargo, esta alineación es muy compleja de conseguir. La falta de esta alineación reduce la efectividad tanto de los sistemas IT como de los procesos. Para mitigar este efecto, el segundo de nuestros objetivos es favorecer la derivación de los casos de uso del sistema, y las relaciones entre estos, a partir de los procesos del cliente. Para ello, se han definido una serie de patrones que facilitan este paso. Mejorando la alineación entre ambos y la eficiencia final obtenida.

1. Introducción

Los procesos de negocio (PN) han ido adquiriendo cada vez un mayor protagonismo en el seno de las compañías software. Actualmente, estas compañías los están utilizando con diversos propósitos. Por un lado, utilizan herramientas e

implantan estándares que les facilitan la gestión y mejora de sus propios procesos de negocio (los procesos software) [8]. Por otro lado, analizan los procesos de negocio de sus clientes como uno de los primeros artefactos de un proyecto [12].

Nuestras líneas de investigación están encuadradas en ambos campos. La primera de ellas, se centra en reducir el esfuerzo que conlleva la gestión de los procesos software. La segunda, se centra en mejorar la alineación entre los procesos de negocio y los sistemas IT.

A partir de la primera línea de trabajo hemos detectado que, para ser competitivas, las compañías software están implantando nuevas técnicas, metodologías y modelos de negocio. Por ejemplo, están implantando estándares de calidad [8] o distribuyendo el desarrollo [13, 10]. Estas técnicas proporcionan una serie de beneficios [21], como el aumento de la calidad o la disminución de costes, pero también conllevan un incremento del número de actividades de gestión, que además suelen ser altamente repetitivas. Por ejemplo, son necesarias actividades para controlar el proceso o para gestionar a los desarrolladores y a las tareas distribuidas. La realización de estas actividades requiere de un gran esfuerzo. Consecuentemente, los gestores deben dedicar un mayor tiempo a estas actividades, disminuyendo el tiempo dedicado a otras que proporcionan un mayor valor e incrementándose los costes que se pretendían reducir.

Con el objetivo de decrementar el esfuerzo necesario para realizar tales actividades, nos centramos en tratar de automatizar el mayor número posible de ellas. Para ello se ha decidido utilizar un motor de procesos software (BPMS). Este BPMS recibe dos modelos BPMN como entrada: el proceso software a ejecutar y los procesos de negocio correspondientes al sistema para el que se desarrolla el producto software. A

partir del primer modelo, el BPMS automatiza actividades de gestión, como la apertura y cierre de iteraciones, el inicio del desarrollo de un caso de uso (con su asignación, desglose en tareas, estimaciones de esfuerzo, etc.) o la notificación de tareas. El segundo proceso permite adaptar el proceso software al producto a desarrollar. Por ejemplo, identificando cuáles son los diferentes casos de uso que hay que desarrollar. Este motor reduce el esfuerzo que hay que dedicar a las actividades de gestión.

A partir de la segunda línea de trabajo hemos detectado que, cada vez es más necesario que los productos desarrollados se alineen con los procesos de negocio del cliente [6]. Esta alineación permite que los productos proporcionen un mayor rendimiento del negocio [7] y que su tecnología sea utilizada de forma estratégica [20]. Sin embargo, todavía existen dificultades para lograr esta alineación [19]. Además de por problemas de comunicación, la raíz de esta dificultad radica en que las técnicas de especificación de requisitos detallan correctamente las funcionalidades, pero no permiten definir todas las relaciones entre requisitos o entre éstos y el entorno. La pérdida de estas relaciones favorece la aparición de defectos. Los cuales, por ejemplo, pueden influir negativamente a la integración de cada caso de uso con el resto o del nuevo sistema con el resto de sistemas del entorno. Estos defectos o deben ser corregidos o pueden provocar una disminución de la eficacia del sistema y, por tanto, del negocio.

Con el objetivo de mejorar esta alineación, estamos trabajando en la definición de una serie de patrones que permitan la identificación de los casos de uso de un sistema a partir de la descripción (en BPMN) de los procesos de negocio del cliente. El proceso de identificación pone especial atención en mantener presentes las relaciones entre casos de uso, y entre éstos y el entorno. Además, se pretende preservarlas a lo largo de todo el desarrollo. Con ello se mejora la alineación entre los sistemas IT y los procesos de negocio que soportan.

Este artículo está organizado de la siguiente forma: la sección 2 describe las motivaciones de ambas líneas de trabajo. La sección 3 indica el estado actual de ambos trabajos. La sección 4 describe los trabajos relacionados. Finalmente, la sección 5 detalla el estado actual y los trabajos futuros.

2. Motivaciones

2.1. Herramientas de gestión de procesos

La gestión de procesos software es una de las actividades más importantes para garantizar el éxito de los proyectos. Por ello, las compañías están implantando herramientas que faciliten la realización de esta actividad.

Actualmente, se están implantando herramientas de gestión cada vez más completas y que cubren un mayor número de actividades. Así, por ejemplo, Rational Team Concert [17] permite modelar el proceso software a seguir para cada proyecto, gestionar las tareas modeladas, asignar tareas a los desarrolladores o controlar el estado de cada tarea. Otro ejemplo es Specula [11], la cual permite a los gestores definir métricas alineadas con los objetivos de cada proyecto.

Sin embargo, a pesar de que estas herramientas son muy útiles para reducir el esfuerzo empleado en la gestión de procesos, sigue siendo necesario realizar todas las actividades de gestión [3]. Además, algunas de estas actividades (como la asignación de tareas o la gestión de recursos) son altamente repetitivas. Esto implica que los gestores deben dedicar un gran esfuerzo y tiempo a realizar tareas repetitivas. Por lo tanto, resultan esenciales herramientas que no solo den soporte a estas actividades, sino que también automaticen el mayor número posible de ellas. De esta forma, los gestores pueden dedicar un mayor esfuerzo a realizar actividades que proporcionen un mayor valor al cliente.

2.2. Alineación de PN y sistemas IT

Las organizaciones están constantemente evaluando y mejorando sus procesos de negocio para que proporcionen el máximo valor al negocio. Para que este valor sea el máximo posible, es necesario que los sistemas IT que los soportan estén perfectamente alineados con ellos.

Por esta razón, las compañías software especifican los requisitos de los sistemas IT a desarrollar en base a las necesidades del negocio del cliente. Sin embargo, en muchas ocasiones, una vez desarrollados los sistemas, éstos no cubren todas las necesidades o aquéllas cubiertas

no están completas [22]. Uno de los principales problemas por lo que esto viene ocasionado es porque la especificación de los requisitos no da lugar a una representación exacta del negocio y sus necesidades [20]. Técnicas como la especificación de los Casos de Uso, las Historias de Usuario o los Goals, permiten identificar y detallar los requisitos que cumplen dichas necesidades. Sin embargo, no facilitan la identificación y especificación de las relaciones entre requisitos o entre éstos y el entorno, perdiendo parte de la información de dichas necesidades y del negocio [19].

Con el objetivo de minimizar este problema, diversas investigaciones han definido métodos que permiten deducir los requisitos a partir del negocio. Así, en [18] se indica cómo obtener diagramas de casos de uso a partir de procesos de negocio. En [9] indican cómo controlar el nivel de detalle de los casos de uso. En [23] se definen algunos patrones para identificar las relaciones entre casos de uso. No obstante, estos trabajos no permiten identificar todas las posibles relaciones [4], ya que no contemplan el tratamiento de casos de uso en ramas paralelas, sub-procesos, flujos de excepción, etc.

La falta de esta información implica que los requisitos sean desarrollados sin tener en cuenta estas relaciones. Una consecuencia negativa de esto es el aumento de la dificultad de integración entre requisitos. O lo que es peor, del sistema desarrollado con el resto de sistemas y procesos del negocio. Esta falta de integración podría conllevar una disminución de la eficacia del sistema o incluso la necesidad su posterior modificación para mejorar dicha integración.

Esto pone de manifiesto la necesidad de métodos que permitan identificar las relaciones entre requisitos y entre requisitos y el entorno. Con este objetivo, se propone el uso de una serie de patrones que, aplicados sobre los procesos de negocio, facilitan la identificación de esta información. De esta forma se contribuye a mejorar la alineación de los sistemas con el negocio, aumentando la eficacia de ambos.

3. Uso de técnicas BPM para mejorar la gestión y alineación de procesos

A continuación se detallan las técnicas BPM (Business Process Management) utilizadas para

automatizar el mayor número posible de actividades de gestión del proceso software y para mejorar la alineación de los productos con el negocio del cliente.

3.1. Automatizando la gestión de procesos

Para poder automatizar las actividades de gestión es necesario saber el estado exacto de cada proceso, qué tareas se están haciendo y cuáles son las que se deben realizar a continuación.

Estas necesidades son en parte ya cubiertas por los BPMS. Los BPMS son motores que ejecutan procesos de negocio, normalmente modelados en BPMN o BPEL, para orquestrar y realizar las tareas definidas en ellos.

Por lo tanto, se decidió utilizar estos BPMS para ejecutar modelos de procesos software y, así, cubrir las necesidades anteriores. Para ello, tanto la notación BPMN utilizada para modelar los procesos software, como el propio BPMS, tuvieron que ser adaptados para soportar las características específicas de la ejecución de estos procesos. Así, en primer lugar se decidió utilizar un BPMS libre, Apache ODE [1], para poder adaptarlo a éstas características sin necesidad de tener que desarrollar uno nuevo. En segundo lugar, se añadieron nuevas funcionalidades al BPMS y nuevos atributos a la especificación BPMN para incluir información específica del proceso software. Algunos de estos atributos son:

- Artefactos y modelos que deben ser generados como resultados de cada tarea.
- Herramientas o funcionalidades que deben ser utilizadas para completar cada tarea.
- Actividad automática o manual. Las actividades automáticas son aquellas que pueden ser completadas sin necesidad de intervención de los usuarios. Las actividades manuales son aquellas en las que al menos un usuario debe trabajar.

A pesar de estas adaptaciones, el funcionamiento de este BPMS es similar al de cualquier otro motor de procesos. Así, cuando el jefe de un proyecto debe decidir qué proceso software seguir, lo modela en BPMN o lo elige de entre los ya modelados. A continuación, se despliega una nueva instancia del proceso elegido en el BPMS. A partir de dicha instancia, el

proceso es monitorizado y, a medida que el proyecto avanza, las tareas definidas como manuales son añadidas a una herramienta de gestión de proyectos, para que sean estimadas y asignadas. Aquéllas definidas como automáticas, son completadas por el BPMS reutilizando información ya insertada. Algunas actividades ya automatizadas son: la apertura y cierre de iteraciones, notificación de cambios de estado o cierre de requisitos ya finalizados.

A pesar de que los proyectos desarrollados siguen el mismo proceso software (o similares), a medida que el proyecto progresa, su desarrollo ha de adaptarse a las características del producto que se desarrolla. Para obtener dicha flexibilidad, este BPMS también ha sido adaptado para que reciba una segunda entrada. Esta entrada recibe los procesos de negocio del producto a desarrollar, o los diferentes artefactos generados a lo largo del ciclo de vida del proyecto. Estos artefactos son utilizados por el BPMS para ejecutar diferentes subprocesos software dependiendo de la información recibida. Así, por ejemplo, cuando el BPMS recibe los procesos de negocio del producto a desarrollar, se ejecuta el subproceso encargado de identificar los casos de uso, y las relaciones entre casos de uso, a partir de dichos procesos. Además, este subproceso gestiona las tareas a realizar para identificar dicha información.

Un beneficio adicional de ejecutar, con un BPMS, el proceso software como un proceso de negocio es el hecho de que, para cada proyecto, la compañía de desarrollo puede poner en marcha diferentes procesos software dependiendo de requisitos tales como la agilidad requerida, el nivel de control de calidad o la tecnología con la que se va a desarrollar. Para ello, simplemente es necesario contar con la descripción BPMN relativa a cada proceso.

3.2. Alineando los sistemas IT con el negocio

Normalmente, los procesos software definen actividades encaminadas a tener en consideración el negocio de las organizaciones, tales como definir los objetivos del negocio o los casos de uso de negocio [12]. Sin embargo, estas actividades están orientadas a exponer el negocio como punto de partida, y no a conseguir una alineación PN-IT. Ello conlleva que en numerosas ocasiones, el

sistema IT desarrollado no se alinee correctamente con el negocio para el que fue desarrollado.

En esta línea de trabajo se propone modelar los procesos de negocio de una organización (utilizando la notación BPMN) como punto inicial para identificar los requisitos del sistema a desarrollar. El objetivo no es solo obtener los requisitos sino mantener presentes las relaciones entre ellos. Para facilitar esta identificación, se han definido una serie de patrones que permiten identificar tanto los casos de uso como las relaciones entre casos de uso (CUs) a partir de los procesos de negocio.

Para definir estos patrones se ha adoptado el concepto de “Step” introducido en [9] (donde se indica que un “step” es la secuencia de tareas que pueden ser realizadas sin interrupción por un actor y, por lo tanto, un “step” es un caso de uso).

Tabla 1. Identificación de casos de uso a partir de procesos de negocio. Ampliado en [4]

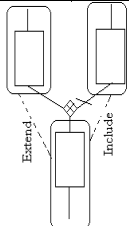
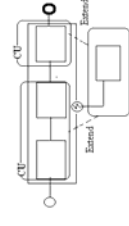
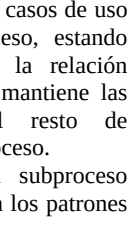
Paso	Descripción
1	Se comienza en el primer evento o actividad del proceso de negocio
2	Todas las actividades conectadas mediante un flujo de control son un CU.
3	Un caso de uso acaba cuando:
	A. El flujo pasa de un actor a otro
	B. Existe una transición de tiempo entre actividades
	C. Se llega al final del proceso

La aplicación de estos patrones se basa en la realización de los siguientes pasos:

- Identificar los actores del sistema. Un actor será: todo agente externo al negocio (pools del proceso), los agentes internos del negocio (pools y lanes) o sistemas heredados.
- Marcar las actividades de los procesos que deben estar soportadas por el sistema, marcando cada actividad (como se indica en [9]): como A (Automática), S (Soportada) o M (Manual)
- Identificar los CUs a partir de los procesos de negocio. Para lograrlo, se deben detectar los conjuntos de tareas que pueden ser realizadas por un actor sin interrupciones, siguiendo los pasos indicados en la tabla 1 (basados en el concepto de “Step” introducido en [9]).

- A partir de este punto, se deben identificar las relaciones entre CUs mediante los patrones definidos. La Tabla 2 muestra algunos de estos patrones. Una extensión de los mismos puede encontrarse en [4].

Tabla 2. Patrones para la identificación de relaciones entre casos de uso

Descripción	Diagrama
Si existen varios CUs unidos por un Gateway de tipo “exclusivo” o “inclusivo”, el CU en la ramificación por defecto se relacionará con el CU inicial mediante “Include” (puesto que éste es el CU por defecto a seguir), y el CU en la rama alternativa mediante “Extends”, puesto que éste sólo se realizará cuando se cumpla su condición.	
<i>Tratamiento de excepciones.</i> Si una excepción cubre varios casos de uso, se crea un nuevo CU para el flujo de excepción. Éste estará relacionado mediante “Extend” con los CUs cubiertos por el flujo excepción. De esta forma el control y tratamiento de la excepción estará encapsulado en un CU.	
<i>Tratamiento de subprocesos.</i> Para cada subproceso se crea un caso de uso de alto nivel que englobe y controle a todos los casos de uso identificados dentro del subproceso, estando relacionado con ellos mediante la relación “Include”. Mediante este CU se mantiene las relaciones existentes entre el resto de actividades del proceso y el subproceso. Los CUs existentes dentro del subproceso estarán relacionados entre sí según los patrones definidos.	

A través de estos pasos y patrones se facilita la identificación de los requisitos del sistema a desarrollar y de las relaciones entre requisitos, o de éstos con otros elementos del entorno (como sistemas heredados). Las relaciones se mantienen presentes durante todo el proceso de desarrollo para conseguir mejorar la alineación e integración entre el negocio y los sistemas IT.

4. Trabajos relacionados

4.1. Herramientas de gestión de procesos

A medida que las compañías software han implantado nuevos estándares o modelos de negocio, han visto como sus procesos se han sobrecargado con actividades de gestión. Esto ha conllevado a que cada vez sean más necesarias herramientas que den soporte a estas actividades.

Orientadas en este sentido se han empezado a desarrollar herramientas que proporcionan un mayor control de los procesos, llamadas software cockpit [16]. Ejemplos de estas herramientas pueden ser Rational Team Concert [17] o Artemis 7 [2]. Estas herramientas dan soporte a la gestión de diferentes procesos software y entornos de desarrollo. Además, facilitan la realización de un gran conjunto de actividades, como el control de asignaciones o la generación de métricas. Sin embargo, sólo dan soporte a la realización de estas actividades, no las automatizan.

Por otro lado, se están realizando investigaciones orientadas a mejorar este tipo de herramientas. Un ejemplo de estas investigaciones es Specula [11]. Esta investigación se centra en la definición de una metodología y de una herramienta capaz de definir componentes reusables para la gestión de procesos. Así, cada compañía puede definir sus componentes y reutilizarlos según sus objetivos y los objetivos de cada proyecto. Sin embargo, al igual que ocurría con las herramientas anteriores, estos componentes sólo dan soporte a la realización de actividades de gestión, no las automatizan.

A diferencia de los trabajos mencionados, una de las líneas de trabajo presentadas en este artículo se centra en el desarrollo de un motor de procesos capaz de automatizar un gran número de actividades de gestión. No existe en estos momentos ningún motor con dichas características.

Finalmente, actualmente existen trabajos orientados a facilitar la ejecución de procesos de negocio, como [5, 24], así como para aumentar la variabilidad y adaptabilidad de los mismos [14, 15]. Estos trabajos están orientados a mejorar estas características para cualquier proceso de negocio. Nosotros nos centramos únicamente en poder ejecutar los procesos software y soportar

cierta adaptabilidad de los mismos con respecto a la situación de los proyectos software.

4.2. Alineación sistemas IT - PN

A medida que las empresas han puesto un mayor énfasis en la necesidad de sistemas IT alineados con su negocio, el número de investigaciones y perspectivas con las que atacar este objetivo se ha incrementado.

Así, en [9] indican una serie de mapeos para obtener actores, casos de uso o, incluso, algunas relaciones entre casos de uso a partir de procesos de negocio modelados con Diagramas de Actividad de UML. Sin embargo, sólo indican cómo obtener relaciones de tipo “extend”. No permiten capturar todas las relaciones existentes (como las relaciones de tipo “include”, casos de uso en sub-procesos, flujos de excepción, etc.).

En [23], extienden los mapeos indicados en [9] añadiendo tres patrones para la identificación de las relaciones entre casos de uso. No obstante, siguen sin contemplar todas las posibles opciones para la identificación de las relaciones (como pueden ser, casos de uso en ramas paralelas, sub-procesos, flujos de excepción, etc.).

Por último, en [18] extienden los diagramas BPMN y los Diagramas de Actividad de UML para que puedan representar procesos de negocios con conceptos de seguridad. Además, presentan una serie de transformaciones en QVT y reglas para obtener los casos de uso, y los casos de uso que proporciona la seguridad, a partir de los procesos de negocio. Sin embargo, no se indica cómo obtener las relaciones entre casos de uso.

A diferencia de los trabajos mencionados, la segunda línea de investigación presentada en este artículo define una serie de patrones que no sólo facilitan la identificación de casos de uso, sino que además permiten identificar las relaciones entre éstos o de éstos con el entorno, a partir de procesos de negocio.

5. Estado actual y trabajos futuros

En este artículo se han presentado dos líneas de trabajo que utilizan técnicas BPM para mejorar tanto la gestión de los procesos de las compañías software como la eficacia de los productos que éstas desarrollan.

La primera línea de investigación está enfocada hacia la agilización de la gestión de los procesos software. Con este fin, se ha adaptado un BPMS para que sea capaz de ejecutar procesos software. Esta ejecución permite automatizar un gran número de actividades, tales como el control de las iteraciones, el control de cambios de estado o el control de los casos de uso. De esta forma, se reduce el esfuerzo dedicado a la gestión de procesos, permitiendo a los gestores centrarse en actividades que proporcionan más valor al cliente. Como trabajo futuro, estamos desarrollando nuevas funcionalidades que incrementan el número de actividades automatizadas, así como la capacidad de adaptación de los procesos software a las características del producto que se desarrolla.

La segunda línea de trabajo se orienta hacia la mejora de la alineación de los sistemas IT con los procesos de negocio del cliente. Como resultado de este trabajo, se han definido una serie de patrones que permiten identificar los casos de uso de un sistema, junto con las relaciones entre éstos, a partir de sus procesos de negocio. Por ejemplo, se han definido patrones para identificar relaciones de tipo extends e include o para el tratamiento de subprocesos. Estos patrones mejoran la alineación IT- Procesos de negocio, mejorando el rendimiento e integración de dichos sistemas con el negocio. Como trabajo futuro, estamos definiendo nuevos patrones que permitan identificar un mayor conocimiento del negocio. Además, se pretende extender un lenguaje de definición de requisitos para que dicha información pueda ser modelada y mantenida a lo largo de todo el desarrollo.

Finalmente, se pretende utilizar la información extraída del negocio para facilitar el desarrollo de sistemas IT. En concreto, de los sistemas desarrollados con una arquitectura multicapa y utilizando frameworks de desarrollo. La construcción de este tipo de aplicaciones requiere de un gran conocimiento. Por ello, se están definiendo reglas que, reutilizando dicha información, faciliten su implementación.

Agradecimientos

Este trabajo ha sido financiado por PDT08A034, TIN2008-02985, Fondos FEDER, PRE09156 y la Fundación Valhondo Calaff.

Referencias

- [1] Apache ODE. <http://ode.apache.org/>
- [2] Artemis 7. <http://us.aisc.com>
- [3] Berrocal, J., García-A., J., Murillo, J.M.: Lean Management of Software Processes and Factories using Business Process Modeling Techniques. PROFES 2010, pp 321-335
- [4] Berrocal, J., García-A., J., Murillo, J.M.: Patrones para la Extracción de Casos de Uso a partir de Procesos de Negocio. PNIS 2009, pp 1-11 (2009)
- [5] Cánovas, J.L., Sánchez, O., García, J.J., Castillo, C.: Un caso de estudio para la adopción de un BPMS. Taller de Procesos de Negocio e Ingeniería Software (2007)
- [6] Chan Y, Huff S, Barclay D, Copeland D (1997) Business strategic orientation, information systems orientation and strategic alignment. Information System Research Vol. 8, No. 2, pp. 125-150
- [7] Cragg P, King M, Hussin H (2002) IT alignment and firm performance in small manufacturing firms. Journal of strategic information systems. Vol. 11, pp. 109-132.
- [8] CMMI. <http://www.sei.cmu.edu/cmmi/>
- [9] Dijkman, R.M and Joosten, Stef (2002) Deriving Use Case Diagrams from Business Process Models. CTIT Technical Reports Series, 08 (02). ISSN 1381-3625
- [10] Greenfield, J., Short, K., Cook, S. and Kent, S.: Software Factories: Assembling Applications with Patterns, Models, Frameworks, and Tools. (2004)
- [11] Heidrich, J, Münch, J.: Goal-Oriented Setup and Usage of Custom-Tailored Software Cockpits. Product-focused software process improvement (2008)
- [12] Jacobson, I., Booch, G., Rumbaugh, J. (1999) The Unified Software Development Process. Addison-Wesley Object Technology Series. ISBN:0-201-57169-2
- [13] Johnson, James R.: The Software Factory: Managing Software Development and Maintenance. John Wiley & Sons. ISBN-10: 047157225X (1991)
- [14] La Rosa, M., Gottschalk, F., Dumas, M., van der Aalst, W.: Linking Domain Models and Process Models for Reference Model Configuration. 10th Workshop on Reference Modeling, Brisbane, Australia (2007)
- [15] Montero, I., Peña, J., Ruiz-Cortés, A.: Business Family Engineering: Does it make sense? Primer taller de Procesos de Negocio e Ingeniería Software (2007)
- [16] Münch, J., Heidrich, J.: Software Project Control Centers: Concepts and Approaches. Journal of Systems and Software 70(1), 3-19 (2003)
- [17] Rational Team Concert. <http://www-01.ibm.com/software/awdtools/rtc/>
- [18] Rodríguez, A.; Fernández-Medina, E; Piattini, M. (2008): Towards Obtaining Analysis-Level Class and Use Case Diagrams from Business Process Models. ER Workshops 2008
- [19] Salinesi C, Thevenet L (2007) Aligning IS to organization's strategy: the INSTAL method. International Conference on Advanced Information Systems Engineering, Norway
- [20] Sase Narine Singh Carson Woo, (2009). Investigating business-IT alignment through multi-disciplinary goal concepts. Requirements Engineering Journal Vol. 14, No. 3 pp. 177-207
- [21] Sengupta, B., Chandra, S. and Sinha, V.: A research agenda for distributed software development. (ICSE 06) Proceeding of the International Conference on Software Engineering, Shanghai, China, May 20-28 of 2006, pp 731-740. (2006)
- [22] Standish-Group (1994) The Chaos report. In: The Standish Group
- [23] Svatopluk Stolf, Ivo Vondrák (2005) Mapping from Business Processes to Requirements Specification.
- [24] Vanderfeesten, I.T.P., Reijers, H.A., van der Aalst, W. Product Based Workflow Support: A Recommendation Service for Dynamic Workflow Execution. 20th International Conference on Advanced Information Systems Engineering (CAiSE'08), volume 5074 of Lecture Notes in Computer Science, pages 571-574. Springer-Verlag, Berlin, 2008.

BP Research Lines at PROS

Victoria Torres

Research Center on Software
Production Methods
Universidad Politécnica de Valencia
46022 Valencia
vtorres@dsic.upv.es

Pau Giner

Research Center on Software
Production Methods
Universidad Politécnica de Valencia
46022 Valencia
pginer@dsic.upv.es

Vicente Pelechano

Research Center on Software
Production Methods
Universidad Politécnica de Valencia
46022 Valencia
pele@dsic.upv.es

Abstract

This paper provides an overview of the past, current and future research interest of the ProS research center focusing on the Business Process field.

1. Introduction

The *Research Center on Software Production Methods* (ProS) has always been interested in the development and application of techniques that improve (in terms of time, quality, and productivity) the construction of software systems. To achieve these improvements, we proposed the use of Model Driven Engineering techniques. These techniques allowed us to specify systems independently of technological details and to (semi)-automatically generate software applications by means of model transformations. Initially, in the early 90s, the research results were successfully applied to the development of information systems [10]. Since then, we have applied these results for developing web applications and managing smart homes. However, in the last years, we are being witnesses of how software applications are transcending the barriers of traditional desktop applications. As a result, nowadays we can find many ubiquitous applications that are integrated into the user environment, by means of mobile or embedded devices, to support her daily activities (e.g. to control domestic heating, to check flight schedules or to book accommodation). However, this type of applications are highly complex to develop since they run in dynamic environments, require a high degree of autonomy and adaptation to context, and require new kinds of interaction mechanisms to support everyday tasks. In this context, the ProS is working on research lines that contribute to solve some of the challenges that introduce these type of

applications. Specifically, focused on the Business Process Management (BPM) field, the PROS is interested in the role played by business process models in this type of applications.

2. Background

ProS members have a strong background in the development of methods for the construction of software applications. Related to the Business Process field, the centre is conducting a research line dedicated to align business processes with IT applications and to solve the challenges that introduce this alignment. Initially, the research work was focused on the development of Web applications from business process models. For this purpose, business process models were introduced within the software development process as main artefacts. The knowledge contained in these models were used, jointly with other models, to build new models (i.e. navigational and presentation models) and to derive (semi)-automatically the corresponding software application in terms of a specific technology. This was possible since the development process was designed using Model Driven Engineering techniques. The whole system was represented by means of models and by the application of model-to-model and model-to-text transformations we could derive the specified system into an executable code implemented in a particular technology. Another challenge faced in this line was related to the integration of the physical and virtual worlds. Information Systems can be aware of physical objects thanks to Automatic Identification (Auto-ID) technologies such as Radio Frequency Identification (RFID). However, due to the technological heterogeneity in Auto-ID and the fast-changing requirements of business processes, there was a need to move from ad-hoc solutions to sound development methods in order to assure the quality of the final product.

Therefore, considering the specific requirements of the linkage between physical and virtual worlds, this work resulted in the definition of a development process that covers from the specification to the implementation of this type of systems. Again, this work was based on Model Driven engineering foundations, what allowed us to systematize the process to improve their construction, maintenance and evolution. Another research line not directly related to business processes but whose results are being applied in this field is related to the Autonomic Computing field. In this case, considering the emerging necessity of adaptation in highly dynamic systems such as context-aware or ubiquitous systems, the research challenges that were faced are to provide new guidelines, techniques and tools to help autonomic system development. To handle variability at run-time this work provided a Model-Based Reconfiguration Engine (MoRE). Given a context event, MoRE queries the variability models to determine how the system should evolve, and then it provides the mechanisms for modifying the system architecture accordingly. To successfully move towards implementing the key self-management properties associated with autonomic computing, this work combined Model Driven Engineering and Software Product Lines techniques.

3. Current Research

There are several research lines in which business process modelling and execution are considered. In this section we provide an overview of the specific aspects in which we are interested in.

3.1. Service and Business Process Variability

In some domains, applications are governed by business processes to achieve a specific goal. Considering a dynamic context where applications should reconfigure and adapt to satisfy new requirements, application changes should be made according to their underlying business processes.

Therefore, in this research line we are interested in dealing with variability issues in business processes. In order to properly deal with these issues we have to consider variability from two complementary points of view, the modelling and execution point of view. On the

one hand, from the modelling point of view, it is necessary to provide techniques that help and assist the business analyst during the definition of business process models. Business process modelling has been recognized as a non-trivial task and its results strongly depend on the skills of the modeller [9]. The challenges that arise at this point relate to the definition of reference process models and their variants:

- Identification of the common elements of a process model
- Identification of the process parts that vary depending on the context.
- Identification of the different alternatives (variants) that fit in each “whole” of the process.
- Deciding the granularity of the variants.

The way in which these challenges are faced will determine other aspects such as (1) model and fragments reusability, (2) model scalability or (3) model change propagation. An important aspect to take into account during the modelling phase is that this is going to be performed by business analysts. This type of stakeholders has a deep knowledge on the domain (business) but at the same time are not experts on software modelling. For this reason, it is necessary to provide languages and techniques that are close to their knowledge.

Within the business process modelling area, several works have been developed to deal with variability modelling. These solutions either (1) extend existing languages, such as EPC [12] or BPMN [11] to explicitly distinguish between normal branching and variability at design time or (2) define mechanisms to model separately the base case from the possible variants [6]. Obviously, the former solution introduces complexity to the model in terms of size (all the possible variants are modelled within the same diagram), reusability (same variants have to be included each time in the diagram) and maintainability (changes in a variant that is used more than once has to be propagated accordingly). Regarding the latter solution, even though it solves the drawbacks found in the former, it still shows limitations when defining variants. For instance, it is not possible to reuse variants within own the definition of variants.

On the other hand, from the execution point of view, business process models are seen as a composition of services that should change according to the current context and the variants that were previously identified during the modelling phase. The challenges that arise in this case are:

- The reconfiguration of already running process instances
- Guaranteeing the correctness of the new composition.

There are several works that deal with variability at the execution level. In the literature we find for instance VxBPEL [8], an extension to the WS-BPEL language to deal with variability. However, this solution just allows managing variability at run-time that has been previously considered at design time. On the contrary, we are interested in achieving business process reconfiguration when new variants (variants that were not considered during design time) appear.

To face these problems, those arising at the modelling and execution level, we propose to support the business process changing nature by the application of Model Driven Engineering techniques. At the modelling level we propose the use of a DSL to properly specify BP variability (in particular we propose the use of CVL (Common Variability Language)). At the execution level we propose the use of business process models at run-time to decide which WS-BPEL configuration is the most adequate at each moment. In addition, MDE techniques can be also used to move from business process modelling to execution. These techniques would allow us abstracting all the complexity required by the underlying technology (e.g. WS-BPEL and Web Services) to successfully implement process variability.

3.2. Physical Mobile Workflows

Business processes in organizations usually involve real-world objects such as baggage in an airport or books in a library. However, there is a gap between the real world where things happen and the digital world where information is handled. The Internet of Things (IoT) vision [4] is about reducing this gap between the physical and the digital world in order to make daily activities more fluent. For example, a librarian can access

the book-related services just by approaching his/her mobile device to the book.

Existing business process management solutions are mainly focused on the digital world, providing orchestration capabilities for Web Services (e.g., based on WS-BPEL). Automatic identification (Auto-ID) technologies such as RFID have been successfully applied for long to automate workflows in different areas such as manufacturing and logistics, retail, animal tracking, and transport and admission ticketing [14]. Different proposals exist with the goal of linking the physical and virtual worlds at the technical level. Proposals such as [3] [7] [13] provide middleware that automatically processes identification events. However, the way in which the physical-virtual linkage has not only impact at the technological level but also at process design. We believe that it is important to describe the physical-virtual linkage for a workflow at design time since the way in which a business goal is achieved depends on the properties of the physical-virtual integration. Certain business models are only feasible with an adequate level of automation in the physical-virtual linkage [2]. For example, using RFID for identifying products in a supermarket allows checkout to be automated, and does not require the participation of a cashier in the process. Thus, when modelling a business process it is not possible to determine which tasks are required for handling physical elements (e.g., requiring a cashier to handle them or not) if there is no notion of how they participate in the process. Models are key in our proposal to provide this notion by linking identification requirements to technological requirements in a gradual manner. In this context, we have been developed a model-based architecture named Presto [5] to capture the concepts that are involved in the physical-virtual linkage for business processes, and derive a software solution by means of Model Driven Engineering techniques (currently generation is supported for the Android platform). The resulting system is capable of presenting to each participant in the process the services that are associated with their physical environment according to their role in the process and the current pending tasks.

4. Future Research

Regarding the future research related to business processes, we are planning to (1) apply the results obtained in the business process variability research line to the Method Engineering research line (see section 4.1) and (2) improve the coordination between embedded services and users to provide workflow driven services in an adequate level of obtrusiveness (see section 4.2).

4.1. Method Engineering

Within the context of this research line, we are interested in giving support to the limitations found in the Situational Method Engineering (SME). This discipline deals with the adaption of Software Production Methods (SPMs) and tools to specific Information System Development projects. Adaptation is required from two different perspectives which are the *product* and the *process* of the SPM. Focusing on the process perspective, we are planning to apply the research results obtained in the BP Variability line to the current challenges of the SME. These challenges are (1) the proper definition of process variability at design time (within a CAME environment) and, (2) the reconfiguration of the SPM supporting CASE tool based on the process (and variability) definition. Therefore, the main outcome expected from this research line is, in a first stage, a methodological framework for the definition of SPMs and for the automatic construction of their supporting tools and then, in a second stage, we plan to enrich this framework to deal with the required adaption of SPMs and tools according to the current application context (e.g. project resources or time constraints).

4.2. Obtrusiveness Orchestration

Service orchestration is normally performed in terms of functionality. Nevertheless, in practice we found that in our experimentation with physical mobile workflows most of the complaints were not related to the functionality provided but the way it was provided instead. In particular we are carrying out research in orchestrating the attention resources of the user. Imagine a future in which your fridge announces to you the recipes that can be prepared with the available goods, your TV tells you that your

favourite program is beginning, the book you want to start reading is suggesting you try other similar books; and all of this is happening at the same time. Clearly living in such an environment on a daily basis would be annoying. On the other hand, if services behave in a completely automatic manner (without requiring human input), users can feel that their environment is out of their control, which is also undesirable.

Since user attention is a valuable but limited resource, we believe that an environment full of embedded services must behave in a considerate manner, demanding user attention only when it is required. In the same way that a musical orchestra requires a conductor to indicate who should play and the tempo to be followed, we propose to incorporate orchestration techniques in smart environments, in order to achieve a balance between automation and user participation. We are currently exploring the use of orchestration techniques and model-based reconfiguration [1] to provide the services that are part of a workflow at an adequate level of obtrusiveness (e.g., making a mobile device rang only if the message relevance and the context is appropriate for doing so).

5. Conclusions

This short paper has provided an overview over the past, current and future research interests of the ProS members in business process field. Nowadays, the challenges associated to the presented lines constitute very hot topics in the business process field. In addition, the background and experience of the ProS members in business process modelling and execution, model driven engineering and system variability constitute a proper context to face the presented challenges.

References

- [1] Cetina, C.; Giner, P.; Fons, J. & Pelechano, V. Autonomic Computing through Reuse of Variability Models at Runtime: The Case of Smart Homes *Computer, IEEE Computer Society*, **2009**, 42, 37-43
- [2] Fano, A. & Gershman, A. The future of business services in the age of ubiquitous computing *Commun. ACM, ACM*, **2002**, 45, 83-87

- [3] Floerkemeier, C.; Roduner, C. & Lampe, M. RFID Application Development with the Accada Middleware Platform *IEEE Systems Journal, Special Issue on RFID Technology*, **2007**
- [4] Gershenfeld, N.; Krikorian, R. & Cohen, D. The Internet of Things *Scientific American*, **2004**, 291, 46-51
- [5] Giner, P.; Cetina, C.; Fons, J. & Pelechano, V. Developing Mobile Workflow Support in the Internet of Things *IEEE Pervasive Computing, IEEE Computer Society*, **2010**, 9, 18-26
- [6] Hallerbach, A.; Bauer, T. & Reichert, M. Capturing Variability in Business Process Models: The Provop Approach *Software Process: Improvement and Practice, Wiley InterScience*, **2010**
- [7] Kindberg, T.; Barton, J. J.; Morgan, J.; Becker, G.; Caswell, D.; Debaty, P.; Gopal, G.; Frid, M.; Krishnan, V.; Morris, H.; Schettino, J.; Serra, B. & Spasojevic, M. People, Places, Things: Web Presence for the Real World. *MONET*, **2002**, 7, 365-37
- [8] Koning, M.; ai Sun, C.; Sinnema, M. & Avgeriou, P. VxBPEL: Supporting variability for Web services in BPEL *Information and Software Technology*, **2009**, 51, 258 - 269
- [9] Mendling, J.; Reijers, H. A. & van der Aalst, W. M. P. Seven process modeling guidelines (7PMG). *Information & Software Technology*, **2010**, 52, 127-136
- [10] Pastor, O. & Molina, J. C. Model-Driven Architecture in Practice: A Software Production Environment Based on Conceptual Modeling *Springer-Verlag New York, Inc.*, **2007**
- [11] Puhlmann, F.; Schnieders, A.; Weiland, J. & Weske, M. Variability Mechanisms for Process Models. Technical Report 17/2005. *Hasso-Plattner-Institut, Postdam*, **2006**
- [12] Rosemann, M., van der Aalst, W.M.P.: A configurable reference modelling language. *Inf. Syst.* 32(1) (2007) 1-23
- [13] Römer, K.; Schoch, T.; Mattern, F. & Dübendorfer, T. Smart identification frameworks for ubiquitous computing applications *Wirel. Netw., Kluwer Academic Publishers*, **2004**, 10, 689-700
- [14] Sheng, Q. Z.; Li, X. & Zeadally, S. Enabling Next-Generation RFID Applications: Solutions and Challenges *Computer, IEEE Computer Society Press*, **2008**, 41, 21-28

Hints on how to face business process compliance *

Cristina Cabanillas, Manuel Resinas and Antonio Ruiz-Cortés

Departamento de Lenguajes y Sistemas Informáticos

ETS Ingeniería Informática

Universidad de Sevilla

41012 Sevilla

{cristinacabanillas, resinas, aruiz}@us.es

Abstract

The concept *business process compliance* refers to the degree of conformance between the business processes of an organization and the regulations and rules that govern it. This paper intends to be a starting point for people interested in business process compliance who have no knowledge about how to address compliance checking. We introduce the four most relevant points to be considered before facing the problem and present some hints for those points in the form of a state of the art based on the literature about business process compliance checking. We also state possible future work in the context of business process compliance derived from this study.

Keywords: compliance, business processes, regulations, modelling language, compliance checking.

1 Introduction

Nowadays there is a trend towards changing the traditional design and development of products and services offered by organizations, often focused on software engineering techniques, by methods and techniques that try to eliminate the need of software engineers

on initial stages of the development cycle. In this context, business processes have emerged as an “easy” way to represent the work performed by an organization with the aim of guiding the development of products and the delivery of services. A business process is a sequence of activities that work together to reach a final goal, i.e. they produce a specific product or provide a specific service. They can be modelled with different workflow languages. Furthermore, companies must fulfill a set of rules, from high-level regulations and frameworks that can be applied to as many companies as desired, such as CMMI, ITIL, COBIT and ISO rules, to low-level business rules that emerge and are applied in the specific environment of a company. These rules usually consist of books written in natural language.

In this scenario, ensuring the compliance of business processes with regulations is becoming increasingly important to organizations, since fulfilling the rules gives them a higher level of quality and is an added value to the services they provide.

Dealing with automatic (or semiautomatic) compliance checking is not an easy task, regarding business processes because many elements are involved in their models (activities, data, ...), and with regard to rules because they cannot easily be represented in a process-oriented way, i.e. visually modelled like business processes. We have identified several issues one should consider before addressing compliance checking and we have classified

*This work has been partially supported by the European Commission (FEDER), Spanish Government under the CICYT project SETI (TIN2009-07366); and project P07-TIC-2533 funded by the Andalusian Local Government.

the current state-of-the-art literature according to those aspects in order to find out the most common behaviour and come up with important open challenges that can be tackled in the future in the context of business process compliance checking.

We believe the first thing to do when facing this problem is giving answers to the following questions:

- *When will we check the degree of compliance?*
- *What are we intending to check?*
- *How will we do it?*
- *What languages will we use to model business processes and rules?*

We are introducing only the four most relevant issues, but we could also think of questions related to the level of automation or the visualization of the degree of compliance. Due to space limitations those questions are out of the scope of this paper.

Section 2 answers the questions above from the literature on techniques to check business process compliance. Finally, section 3 draws a set of conclusions from the study and envisages some challenges that can be addressed in the future.

2 How to address business process compliance checking

In the following we present the study we have carried out to answer the questions planned above.

2.1 When will we check the degree of compliance?

The problem of ensuring compliance of business processes with regulations and business rules can be tackled from two main perspectives. At first, researchers opted for a retrospective detection of compliance, i.e. “after-the-fact” or reactive detection, also known as Backward Compliance Checking (BCC). The work in [1, 4, 18, 21] is a good representative of

this approach. BCC techniques’ main flaw is that they can neither prevent the occurrence of non-compliant situations nor modify the behaviour of the process instance during its execution to solve problems, since they just compare the results of the execution with the expected behaviour once the process execution is over.

Forward Compliance Checking (FCC) emerged with a much more preventative focus, with the aim of avoiding the previous problems. FCC techniques target the verification of rules at design time or run time, resulting in two sub-approaches: Design-Time Compliance Checking (DTCC), commonly called “compliance by design” and Run-Time Compliance Checking (RTCC). Checking compliance at design time means that we must try to make the business process comply with the rules since its design, so we can prevent non-compliant situations while modelling both elements. Similarly, RTCC techniques try to check the rules at execution time, which has some advantages, e.g. finding a non-compliant circumstance while running the process may let us solve the problem on time to avoid ending in a non-compliant result. Furthermore, RTCC can check more aspects than DTCC, such as data or resources and performance information, in an easier way. We will further discuss these aspects in the next section.

Most of the current work focuses on FCC approaches, especially on DTCC. The work in [6–15, 20, 22] comprises techniques to check the compliance at design time. Awad et al. [2] use a language called BPMN-Q to check the compliance during the execution of business processes.

2.2 What are we intending to check?

Both business processes and rules range over many aspects we should consider when addressing compliance checking.

- **Control flow.** It is the order in which tasks must be run. Business process modelling languages usually show the control flow implicitly, since modelling business

processes mainly consists of representing the execution order of their activities over time. Therefore, the control flow is inherent in these models.

- **Data.** The execution of business process tasks may involve managing a large amount of data, e.g. information stored in databases may change, new data may be produced and tasks may need specific pieces of data to complete. This information can flow along the process in the form of data objects, e.g. in the form of documents. Rules concerning data management can be defined, so data objects must be also represented in the model and could be the object of compliance checking.
- **Resources.** People interacting with the business process can be considered resources and can be modelled with the process in the form of external agents. Rules stating how they must interact with the process may be defined, so they constitute an aspect that should be taken into account.
- **Time.** Temporal constraints, such as a task expiration date, are handled by means of events in most of business process modelling languages. Rules may impose restrictions on task and process execution timing, e.g. we can indicate that a task must wait eight days to be launched.

Checking the correct work of control flow, data and resource management and the fulfillment of temporal requirements is a non-trivial task that must be carried out when running a business process instance. The fact that a rule can add constraints to business processes on one or more of these aspects makes this checking even more difficult.

The main and most straightforward aspect to check in business process compliance is the control flow. Most of the current techniques assume it and do not specify whether or not they deal with the other aspects.

2.3 How will we do it?

Planning the procedure we will follow is maybe the most important task we must do before tackling the problem. It is strongly related to the classification described in Section 2.1, since techniques may be different depending on when we want to do the compliance checking.

BCC techniques will only be able to decide whether a process instance complied with the rules or not and do nothing to solve non-compliant cases but letting the experts know, so the problems can be solved before the execution of the next business process instance. We can think about monitoring business processes to check the point at which they fail or just compare them with previously run instances. The technique introduced in [18] consists of quantifying how much the execution of a business process matches its expected behaviour by comparing it with previous instances registered in history logs. Its main shortcoming is its inability to deal with data fields, temporal aspects and run-time information, since the checking takes place when the execution is over. The approach described in [21] is based on Linear Temporal Logic (LTL) and its aim is to verify if a given LTL rule holds for a set of process instances, separating the result into two groups: one containing the compliant process instances and one with the non-compliant ones. Although it can handle data and temporal and performance aspects, the inexistence of a graphical notation hinders the work of analysts. In [4] this problem is solved by modelling the rules with a graphical language called GOSpeL. The models are then translated into the declarative language SCIFF and applied over process instances the same way as Alberti et al. [1] do. Also, the approach of Alberti et al. [1] allows both kinds of FCC by means of the language g-SCIFF, developed by the authors.

Regarding FCC, many authors propose to separately model the business process and the rules, do the transformations required to convert the two models into formal representations (see Section 2.4) and then apply model-checking algorithms to evaluate the degree of

compliance [2, 6, 14, 19]. Some of these techniques include the presentation of counterexamples that show where the non-compliance problem has arisen. Sometimes the authors use patterns, either to simplify the modelling of the rules or to facilitate the task of checking compliance in scenarios that often recur. The procedure described in [8] is based on the annotation of business process models with effects derived from the contracts. It requires a process for pair-wise effect accumulation and the semantics necessary to transform the annotated model into a Semantic Process Network (SPNet), which is used to check the degree of compliance. Some structural and semantic patterns are proposed to avoid the occurrence of non-compliant situations. In a similar way, Weber et al. [22] introduce a formalism that acts on business process models in any typical workflow language annotated with predicate logic. A propagation algorithm is used to go along the process and its result is the input for compliance checking. The shortcoming of this approach is that the formalism does not support loops and computationally hard cases and it does not deal with resources and temporal aspects. The idea in [12, 13] is to enhance compliance management by leveraging compliance checking to a semantic level through ontologies. Namiri et al. [16] present a three-step compliance checking approach based on the introduction of a new layer with the representation of controls (i.e. rules) over the business process model. However, this technique requires much manual work by experts. The authors in [10, 11, 15] lean on the idea of accumulating effects across the tasks and use the concept of Ideal Semantics to assess compliance according to the degree of idealism the process results on. The approach described by Goedertier et al. [9] consists of declaratively capture the rules (with their own language called PENELOPE) and (re)use them to generate the business process model that will be checked for compliance. Ghanavati et al. [7] describe a constraint management framework for compliance checking focused on the hospital domain. The approach separately models the functional and opera-

tional aspects of business processes, the non-functional aspects of them and the regulatory policies and then tries to link them in order to find non-compliances. Its main flaws are its low scalability and the great need of manual compliance checking.

2.4 What languages will we use to model business processes and rules?

The term compliance involves two elements. On the one hand are the business processes that respond to enterprise needs and show what activities must be carried out in a company. On the other hand are the rules the company must fulfill. Both elements need to be modelled somehow in order to automatically check the degree of compliance between them, so proper languages must be chosen for that purpose.

Regarding business processes, most of the authors opt for BPMN as modelling language, since it is very intuitive and easy to use [17]. Some of them annotate the BPMN models with other languages that define the rules [8, 10, 11, 19, 22]. This way, the subject of the compliance checking will be a BPMN model enriched with the rules. Liu et al. [14] propose the de facto standard BPEL for business processes modelling. Ghanavati et al. [7] model them with Use Case Maps (UCM). These three notations have a graphical visualization and are supported by existing tools. Other authors prefer not to specify a language and say just that a directed graph or any business process execution language can be used for process modelling [6, 14]. Kharbili et al. [12, 13] say nothing about how to model the processes and focus on developing an ontology for rules.

As BPMN semantics is not well specified, since there is not a standard metamodel, a solution could be translating the BPMN into PNML (Petri Net Markup Language). Dijkman et al. [5] have developed a tool with this aim.

As far as rules are concerned, some approaches use formal languages to model them, such as FCL (Formal Contract Language) [10, 11, 19] and PENELOPE (Process ENtailment from the ELicitation of Obligations and

Permissions) [9], predicate logic [22] and Conjunctive Normal Form (CNF) [8]. The main advantage of this kind of languages is that they can be more easily interpreted by the machine. On the contrary, they do not have a graphical representation, so the effort made by analysts is cumbersome. This is solved by languages such as BPSL (Business Property Specification Language) [14], GRL (Goal-oriented Requirement Language) [7], BPMN-Q [2] and PPSL (Process Pattern Specification Language) [6]. The two last ones are quite similar and very close to each other in terms of expressiveness. The main difference between them is that PPSL models constraints against UML Activity Diagrams and BPMN-Q does not.

Many approaches propose modelling both business processes and rules with graphical languages and then translating them into formal specifications. BPMN and BPEL models are usually translated into FSM (Finite State Machine) and graphical languages for rule definition are usually translated into LTL (Linear Temporal Logic), a widely used language for specifying temporal properties of software and hardware designs. These languages are suitable as input of most of the current model-checking tools.

3 Conclusions

The main intention of this paper was to act as a guide for those interested in facing the automatic (or semiautomatic) checking of the degree of compliance between the business processes and the rules that govern the business of an organization. The conclusions drawn from the four analysed points are the following: (1) Most of the proposed techniques focus exclusively on checking the compliance while designing the process, although sometimes they consider also run-time aspects. Only one of the approaches developed so far can cover both FCC and BCC. However, this approach is incomplete for our purpose because it handles only rules and say nothing about business processes. (2) Only two (control flow and time) out of the four elements that must be checked

are taken into account in most of the approaches. This is partially due to the fact that languages such as LTL allows modelling these two aspects implicitly. (3) Although each approach has its own features that may make it better or worse with respect to others, the most common scenario regarding the procedure used to do the compliance checking consists of separately modelling the processes and the rules (using proper languages), translating these models into more formal languages and applying some model-checking algorithm to measure the degree of compliance. (4) Finally, we have found out that the most commonly used process modelling language is BPMN [17], and that, on the contrary, several languages have emerged in order to represent the compliance rules and they can be quite different from each other, usually depending on the kind of rules the authors have considered.

From this study we can also conclude that business process compliance checking is a complex problem and further research can be done on this topic. For instance, an interesting challenge could be how to model the rules, which is an important gap between processes and regulations. We could also think of including aspects such as data and resources in compliance checking, since they are excluded in most of the current approaches. Some work on data-related compliance problems that can appear in a business process model has been done, but there is no tool that integrates data-aware compliance into any existing compliance checking framework yet [3].

References

- [1] M. Alberti, F. Chesani, M. Gavanelli, E. Lamma, P. Mello, M. Montali, and P. Torroni. Expressing and verifying business contracts with abductive logic programming. *Int. J. Electron. Commerce*, 12(4):9–38, 2008.
- [2] A. Awad, G. Decker, and M. Weske. Efficient compliance checking using bpmn-q and temporal logic. In M. Dumas, M. Reichert, and M.-C. Shan, editors, *BPM*,

- volume 5240 of *Lecture Notes in Computer Science*, pages 326–341. Springer, 2008.
- [3] C. Cabanillas, M. Resinas, and A. Ruiz-Cortés. On the identification of data-related compliance problems in business processes. In *JSWEB*, page in press., 2010.
- [4] F. Chesani, P. Mello, M. Montali, and S. Storari. Testing careflow process execution conformance by translating a graphical language to computational logic. In *Artificial Intelligence in Medicine*, pages 479–488. 2007.
- [5] R. M. Dijkman, M. Dumas, and C. Ouyang. Semantics and analysis of business process models in BPMN. *Inf. Softw. Technol.*, 50(12):1281–1294, 2008.
- [6] A. Forster, G. Engels, T. Schattkowsky, and R. V. D. Straeten. Verification of business process quality constraints based on visual process patterns. In *Theoretical Aspects of Software Engineering, 2007. TASE '07. First Joint IEEE/IFIP Symposium on*, pages 197–208, 2007.
- [7] S. Ghanavati, D. Amyot, and L. Peyton. Towards a framework for tracking legal compliance in healthcare. In *Advanced Information Systems Engineering*, pages 218–232. 2007.
- [8] A. Ghose and G. Koliadis. Auditing business process compliance. In *ICSOC*, pages 169–180, 2007.
- [9] S. Goedertier and J. Vanthienen. Designing compliant business processes from obligations and permissions. In J. Eder and S. Dustdar, editors, *Business Process Management Workshops*, volume 4103 of *Lecture Notes in Computer Science*. Springer Verlag, 2006.
- [10] G. Governatori, Z. Milosevic, and S. W. Sadiq. Compliance checking between business processes and business contracts. In *EDOC*, pages 221–232. IEEE Computer Society, 2006.
- [11] G. Governatori and S. Sadiq. The journey to business process compliance. In *Handbook of Research on BPM*, pages 426–454. IGI Global, 2009.
- [12] M. E. Kharbili and S. Stein. Policy-based semantic compliance checking for business process management. In P. Loos, M. Nuttgens, K. Turowski, and D. Werth, editors, *MobIS Workshops*, volume 420 of *CEUR Workshop Proceedings*, pages 178–192. CEUR-WS.org, 2008.
- [13] M. E. Kharbili, S. Stein, I. Markovic, and E. Pulvermuller. Towards a framework for semantic business process compliance management. Montpellier, France, 2008.
- [14] Y. Liu, S. Muller, and K. Xu. A static compliance-checking framework for business process models. *IBM Systems Journal*, 46(2):335–362, 2007.
- [15] R. Lu, S. Sadiq, and G. Governatori. Compliance aware business process design. In A. H. M. ter Hofstede, B. Benatallah, and H.-Y. Paik, editors, *5th International Conference on Business Process Management (BPM 2007)*, pages 120–131. Springer, 2008.
- [16] K. Namiri and N. Stojanovic. Using control patterns in business processes compliance. In M. Weske, M.-S. Hacid, and C. Godart, editors, *WISE Workshops*, volume 4832 of *Lecture Notes in Computer Science*, pages 178–190. Springer, 2007.
- [17] OMG. Bpmn 2.0 beta 1. Recommendation, OMG, 2009.
- [18] A. Rozinat and W. M. P. van der Aalst. Conformance checking of processes based on monitoring real behavior. *Information Systems*, 33(1):64–95, 2008.
- [19] S. W. Sadiq, G. Governatori, and K. Namiri. Modeling control objectives for business process compliance. In G. Alonso, P. Dadam, and M. Rosemann, editors, *BPM*, volume 4714 of *Lecture*

- Notes in Computer Science*, pages 149–164. Springer, 2007.
- [20] M. Saeki and H. Kaiya. Supporting the elicitation of requirements compliant with regulations. In Z. Bellahsene and M. Leonard, editors, *CAiSE*, volume 5074 of *Lecture Notes in Computer Science*, pages 228–242. Springer, 2008.
- [21] W. M. P. van der Aalst, H. T. de Beer, and B. F. van Dongen. Process mining and verification of properties: An approach based on temporal logic. In *OTM Confederated International Conferences: CoopIS, DOA and ODBASE*, volume 3760, pages 130–147. Springer-Verlag, October 2005.
- [22] I. Weber, G. Governatori, and J. Hoffmann. Approximate compliance checking for annotated process models. Technical report, 2008.

Aplicación del Paradigma de las Redes-en-Redes en los Entornos de Procesos de Negocio

Javier Fabra, Joaquín Ezpeleta

Grupo de Integración de Sistemas Distribuidos y Heterogeneos (GIDHE)

Universidad de Zaragoza

{jfabra,ezpeleta}@unizar.es

Resumen

El paradigma de la Computación Orientada a Servicios (SOC) representa la base para el desarrollo y ejecución de procesos de negocio dinámicos. Sin embargo, los lenguajes y enfoques actuales para la integración de este tipo de procesos no aportan la flexibilidad ni la dinamicidad requerida en los entornos interorganizacionales en los que son frecuentes los cambios en los requisitos y objetivos de negocio. La plataforma DENEb surge como posible solución a este problema de integración, apoyándose en la separación explícita de la lógica de un proceso mediante el enfoque conversacional y en los beneficios de la utilización del paradigma de las Redes-en-Redes, ofreciendo una serie de mecanismos y técnicas que permiten el modelado de procesos flexibles, dinámicos y adaptables. Como resultado, el desarrollo de DENEb ha abierto una serie de líneas de trabajo y de investigación que se resumen en este artículo.

1. Motivación

En los últimos años se ha vivido una creciente actividad en el área de investigación de la Computación Orientada a Servicios (SOC) aplicada al desarrollo de procesos de negocio. Como resultado, SOC se ha centrado en el desarrollo de aplicaciones orientadas, entre otros, al campo de la gestión de procesos de negocio (BPM). SOC basa el desarrollo de estas aplicaciones utilizando los servicios como elementos básicos de primer orden. En términos de modelos de interacción, la interoperabilidad entre componentes se lleva a cabo mediante la descripción de las capacidades de los servicios y sus interacciones. Diversos estándares se han creado para este propósito, destacando aquellos

pertenecientes a las tecnologías de servicios Web (BPEL, WS-CDL o WSDL, por ejemplo) o los estándares basados en semántica (OWL-S, WSMO, WSDL-S o SWSF, entre otros).

Los servicios Web representan la tecnología más popular para tratar de satisfacer los requisitos para la integración de servicios en el desarrollo de procesos de negocio. Sin embargo, el estado actual de estas tecnologías carece de los mecanismos necesarios para satisfacer los nuevos requisitos de flexibilidad y dinamicidad que SOC propone, limitando el desarrollo de las nuevas aplicaciones. En el caso concreto de los lenguajes de descripción de procesos existe una profunda dependencia entre los conceptos de orquestación y coreografía, lo que dificulta la aplicación de técnicas de integración basadas en la simplificación o la separación de los conceptos relacionados con la lógica interna de los procesos. Un claro ejemplo de esto es el lenguaje de ejecución de procesos de negocio, BPEL [7]. Un proceso descrito en BPEL integra la lógica de negocio y la lógica de interacción dentro de la misma descripción, por lo que cualquier cambio que afecte a una de estas lógicas también afecta a todo el proceso.

La próxima generación de servicios Web requiere una mayor flexibilidad y una mejor interoperabilidad para permitir la implementación de aplicaciones mediante la composición de servicios en entornos evolutivos como los escenarios de gestión de procesos de negocio [1]. Un caso particular de estas aplicaciones son los procesos Web, aquéllos que se ejecutan en entornos interorganizacionales y que tienen que ser capaces de adaptarse dinámicamente a las nuevas políticas y estrategias de negocio y de manejar, en tiempo de ejecución, eventos y cambios inesperados [1, 2, 3, 4]. La flexibilidad de la infraestructura tecnológica en la que se basan estos procesos resulta determinante para

establecer la forma en la que una compañía u organización puede enfrentarse a los problemas y escenarios actuales.

Por tanto, parece que es necesario buscar una alternativa que supere las limitaciones impuestas por los estándares actuales, que aporte mecanismos y técnicas para el desarrollo e implementación de procesos flexibles y dinámicos y que permita gestionar los aspectos de composición e interacción de forma independiente de la tecnología. Con estas premisas se inició la línea de investigación a través de la cual se desarrolló la plataforma para el modelado, desarrollo y ejecución de procesos Web dinámicos, DENEb.

2. El Enfoque Conversacional

Internamente, la lógica de un proceso puede separarse entre la *lógica de negocio* y la *lógica de interacción*. La primera especifica el flujo de tareas que deben ser ejecutadas en el proceso de acuerdo a un conjunto de restricciones de orden. Normalmente, una tarea concreta corresponde con la ejecución de una acción interna (por ejemplo, el acceso a un repositorio de datos de la organización propietaria del proceso de negocio o un cómputo utilizando datos locales al propio proceso) o la interacción con una entidad *software* externa (otro proceso de negocio, un servicio Web, un agente *software*, etc.). Estas interacciones externas están descritas por medio de *protocolos de interacción* o *políticas de conversación* [5]. Un protocolo describe fundamentalmente qué mensajes son intercambiados en el marco de una interacción, cuál es la estructura y el contenido de los mensajes intercambiados, en qué orden se envían los mensajes, etc. Además, un protocolo está organizado como una colección de roles, donde cada rol describe la parte del protocolo que debería ser ejecutada por cada proceso participante en la interacción. No obstante, en un protocolo pueden existir varias secuencias válidas de intercambio de mensajes, recibiendo cada una de ellas el nombre de conversación.

Desde el punto de vista de un proceso concreto, las acciones necesarias para ejecutar las distintas conversaciones que puede mantener con otros procesos durante su ejecución reciben el nombre de *lógica de interacción*. Tradicionalmente, en la implementación de los

procesos de negocio están mezcladas la lógica de negocio y la lógica de interacción. Este es el caso de los procesos implementados en el estándar BPEL. Evidentemente, estas soluciones son poco flexibles desde el punto de vista de la integración de procesos. Dado un conjunto de procesos que están colaborando, cualquier modificación en su protocolo de interacción implica detener la ejecución de los procesos involucrados, reprogramar la lógica de interacción de su implementación y, finalmente, volver a ponerlos en ejecución para que colaboren en base al nuevo protocolo.

El enfoque conversacional [5, 6] propone la separación explícita entre los aspectos de la lógica de negocio (los *workflows*) y los aspectos de interacción (los protocolos). El objetivo de esta propuesta de diseño es permitir la ejecución de distintos protocolos de interacción reutilizando la misma lógica de negocio. De esta manera, un proceso será capaz de seleccionar un protocolo de un conjunto predeterminado y ejecutar su correspondiente rol sin necesidad de detener su ejecución para modificar su implementación. No obstante, el enfoque conversacional permite alcanzar una mayor flexibilidad en la integración de los procesos. Por ejemplo, los procesos podrían determinar (e incluso negociar) en tiempo de ejecución el protocolo en base al cual van a colaborar y, una vez consensuado, ejecutar el protocolo para completar la interacción. Para conseguir esta flexibilidad es necesario previamente que los procesos compartan una manera común de describir los protocolos y que los procesos dispongan de mecanismos a nivel de infraestructura para el envío, recepción y ejecución de los protocolos (roles).

3. El paradigma de las Redes-en-Redes

Las redes de Petri [8] se han utilizado intensivamente en la especificación, análisis e implementación tanto de *workflows* [9] como de protocolos de comunicación [10]. La familia de formalismos de las redes de Petri resulta de gran interés, dado que aporta una semántica formal, una notación gráfica muy intuitiva, y una serie de herramientas y técnicas para su análisis, simulación y ejecución. Sin embargo, las redes de Petri ordinarias (también conocidas como redes lugar/transición) tienen una estructura estática que no resulta lo suficientemente flexible para dar

soporte a la flexibilidad requerida en los nuevos entornos de gestión de procesos de negocio y procesos Web. Sin embargo, el paradigma de las Redes-en-Redes [11], perteneciente al formalismo de las redes de Petri objeto, resulta un enfoque muy interesante. El aspecto clave de este paradigma es que formaliza la utilización de las redes en sí como *tokens* de una red de Petri. Esto permite modelar estructuras jerárquicas muy elegantes, simplificando la complejidad de los sistemas. Las Redes-en-Redes se consideran redes de Petri de alto nivel que, principalmente, aportan los conceptos de instancia de red, nuevos tipos de arcos y la utilización de canales síncronos para comunicar las diferentes instancias de las redes entre sí, posibilitando el intercambio de información.

En la línea de investigación encabezada por DENEb se planteó la utilización de las Redes-en-Redes para modelar los aspectos de la lógica de negocio y de la lógica de interacción de los procesos, esto es, la implementación del enfoque conversacional en los entornos BPM. La utilización del mismo formalismo para estas tareas simplifica los aspectos relativos a su integración. Concretamente, las *Reference nets* [11], una implementación de las Nets-within-Nets, se utilizaron para modelar procesos de negocio dinámicos y sus correspondientes protocolos de interacción (por ejemplo, como un lenguaje de composición y coordinación), y también para implementar la plataforma DENEb para la definición, integración y ejecución de procesos. La interoperabilidad de la plataforma queda garantizada, ya que permite a los procesos de negocio internos cooperar de forma transparente con procesos externos heterogéneos (como procesos implementados en base a tecnologías de servicios Web, por ejemplo) [13,14].

4. La plataforma DENEb

La arquitectura de la plataforma DENEb está formada por tres componentes principales, tal y como puede observarse en su Arquitectura de Referencia mostrada en la Figura 1. En primer lugar, el componente llamado *Workflow and protocol enactment service* integra los motores de ejecución tanto de workflows como de protocolos, que encapsulan la separación explícita entre la

lógica de negocio (*workflows*) y la de interacción (protocolos).

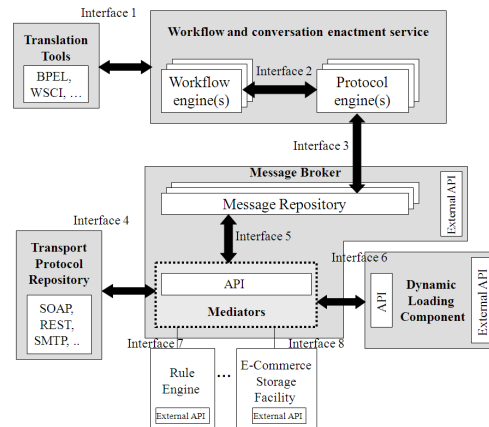


Figura 1. Arquitectura de Referencia de DENEb.

En segundo lugar, el *broker* de mensajes es el componente encargado de la coordinación necesaria durante la ejecución de los procesos. El *broker* está compuesto por un repositorio de mensajes basado en el modelo de coordinación Linda, también implementado mediante las *Reference nets* y denominado DRLinda [15]; y por un subsistema de mediadores que tienen una doble funcionalidad: por una parte, aíslan internamente la plataforma de los detalles concretos de las tecnologías utilizadas; por otra parte permiten extender las características de DENEb fácilmente, al posibilitar el desarrollo de nuevos mediadores en base a un API pre-establecida. Un tipo de mediadores concretos son los responsables de las interacciones con procesos externos utilizando tecnologías concretas, por ejemplo de servicios Web. Igualmente, otro tipo de mediadores son responsables de la realización de procesos complejos que además se exponen públicamente a través de sus interfaces (utilizando WSDL, por ejemplo). Finalmente, existe un tipo concreto de mediadores responsables de dar soporte a los diferentes estándares existentes para la descripción y modelado de procesos. En este caso, estos mediadores representan herramientas de traducción de estos estándares al formalismo de las *Reference nets* utilizado en la implementación de los procesos en DENEb, como el caso del mediador BPEL2DENEb [16].

El último componente de la arquitectura de DENEb está representado por el *Dynamic Loading Component* (DLC). Este componente es responsable de la carga y adquisición dinámica de componentes en la plataforma en tiempo de ejecución. Es posible adquirir y cargar tanto mediadores como *workflows* y protocolos en DENEb, para lo que se posibilitan diferentes alternativas. Esto dota a la plataforma de la capacidad de adaptación a escenarios complejos y evolutivos, mejora la reutilización de la lógica de procesos y abre las puertas de campos de aplicación en desarrollo como la composición dinámica de procesos.

DENEb se basa en la utilización de la herramienta Renew [12], que aporta una interfaz gráfica y un *toolkit* para el modelado de este tipo de redes y que hemos adaptado para permitir el modelado de procesos en base al enfoque conversacional. Internamente, las redes están representadas mediante PNML extendido, lo que resuelve y favorece las cuestiones relativas a interoperabilidad. Además, Renew permite tanto la simulación como la ejecución de las redes modeladas, lo que permite afirmar que tanto los *workflows* como los protocolos desarrollados en DENEb son directamente ejecutables.

5. Dominios de aplicación de DENEb

La utilización del paradigma de las Redes-en-Redes y de las *Reference nets* como tecnología de implementación, tanto para la plataforma DENEb como para los aspectos relacionados con los procesos de negocio posibilitó la aplicación de esta línea de investigación en diversos dominios de aplicación, resumidos en la Figura 2.

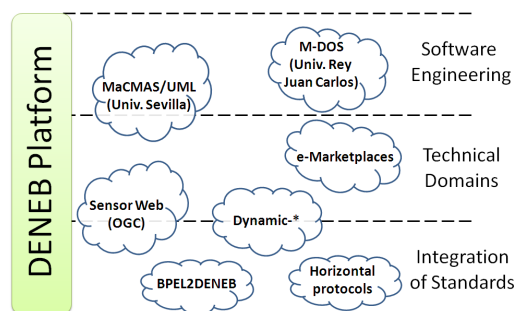


Figura 2. Dominios de aplicación de DENEb.

En lo referente a BPM, las carencias de los estándares actuales para el modelado e implementación de procesos se suplió con la posibilidad de desarrollar dichos procesos utilizando las *Reference nets* y la plataforma DENEb como *frontend* para su modelado y *backend* para su ejecución. La implementación del enfoque conversacional dota de un cierto grado de flexibilidad a dichos procesos, y la utilización de los mecanismos de adquisición dinámica del componente DLC aporta el dinamismo necesario para afrontar con garantías de éxito los nuevos requisitos para el desarrollo de procesos adaptables [13,14]. Además, los modelos generados son directamente ejecutables, lo que favorece la aplicabilidad de DENEb en escenarios de ejecución de modelos.

Relacionado con este último punto se analizó la posibilidad de abordar el desarrollo de procesos de negocio desde la perspectiva de la Ingeniería del *Software*. Se vió que la integración de DENEb con metodologías y plataformas basadas en MDA resultaba sencilla, por lo que diversas colaboraciones dieron su fruto en la integración de DENEb con MaCMAS (*Methodology for Analysing Complex MultiAgent Systems in UML*) [17], permitiendo la generación semiautomática de modelos directamente ejecutables mediante una metodología de refinamientos sucesivos basada en el enfoque conversacional. Por otra parte, la integración de DENEb con la metodología SOD-M [18] aportó la generación automática de modelos ejecutables partiendo de los objetivos de negocio manejados por los analistas mediante modelos de composición de servicio. Estos dos trabajos demuestran la aplicabilidad de la plataforma a diferentes enfoques dentro de la Ingeniería del *Software*, obteniendo idénticos resultados: la reducción de costes en recursos, tiempo y dinero en el desarrollo de procesos de negocio, enriquecida con los beneficios de la generación (semi)automática de código ejecutable.

Las capacidades dinámicas de DENEb también se han explotado en otros entornos de aplicación, como el desarrollo de *e-marketplaces* configurables [19]. En este tipo de escenarios, el desarrollo de protocolos de e-negociación en base a los conceptos y tecnologías desarrollados en DENEb demostró la posibilidad de realizar prototipado rápido y la adaptación de la plataforma de ejecución mediante mecanismos de extensión basados en el DLC. Otra aplicación es

la iniciativa *Sensor Web*, propuesta por el consorcio OGC [21]. En la arquitectura de Sensor Web, las diferentes estaciones que recogen las observaciones de los sensores que abarcan deben acceder a los mismos en base a una serie de protocolos de interacción. Los estándares propuestos por OGC no especifican los detalles de dichos protocolos, ya que normalmente dependen de los fabricantes de los sensores. Las diferentes implementaciones existentes de *Sensor Web* obligan a que estos protocolos sean conocidos en tiempo de diseño y previamente al despliegue del sistema, limitando su alcance y la funcionalidad. La utilización de la plataforma DENEb permite el desarrollo de los procesos que implementan la lógica de negocio de las diferentes estaciones que componen la red de sensores, posibilitando la adquisición de los protocolos de acceso a los sensores mediante el descubrimiento en tiempo de ejecución. Por tanto, el sistema resultante es altamente flexible y escalable, integra los estándares de la OGC y aporta el acceso a los sensores cuyos protocolos de interacción son desconocidos en las etapas de diseño (y que, adicionalmente, pueden utilizar cualquier tecnología para su acceso).

6. Líneas de Investigación Abiertas

Además de los dominios de aplicación de la plataforma, el desarrollo de DENEb y, concretamente, de su componente de adquisición dinámica, ha abierto varias líneas de investigación en la actualidad.

- **Composición dinámica.** El DLC abre nuevos retos en la aplicación del paradigma de las Redes-en-Redes y el enfoque conversacional para desarrollar mecanismos de composición dinámica a nivel de procesos. En este aspecto, en la actualidad se están estudiando los mecanismos existentes y sus limitaciones, y la aplicabilidad de los mecanismos de DENEb para posibilitar el desarrollo de procesos capaces de adaptarse dinámicamente mediante la composición en tiempo de ejecución tanto de su lógica de negocio (sus *workflows*) como de interacción (sus protocolos).
- **Integración con estándares para el modelado de procesos de negocio.** Tan

importante como el desarrollo de DENEb son los detalles para su implantación en entornos de producción reales. Es un hecho que el lenguaje de descripción de procesos más popular es BPEL, pese a sus limitaciones. Por tanto, resulta primordial el estudio de la transformación de este y otros estándares predominantes en este tipo de entornos al formalismo utilizado en nuestra plataforma. Como resultado, esta línea se abrió con la implementación de una herramienta de traducción que permite la utilización de DENEb en entornos empresariales cuyos procesos han sido desarrollados enteramente en BPEL [16]. Los procesos resultantes se implementan mediante *Reference nets* siguiendo el enfoque conversacional de DENEb, aunque no explotan esta característica (semánticamente, el proceso generado es equivalente semánticamente al proceso BPEL original). Sin embargo, en la actualidad se están explotando las ventajas de realizar esta traducción, extendiendo la capacidad expresiva de BPEL en entornos dinámicos. Otra línea abierta es el estudio de la correlación entre los modelos DENEb y BPMN 2.0, dado que este último es el otro estándar más destacado de definición de procesos de negocio.

- **Semántica.** La característica de adquisición dinámica en de workflows y protocolos abre la línea de investigación correspondiente al estudio de la verificación de sus propiedades. Esto se resume en un problema de compatibilidad entre *workflows* y protocolos que se ha estudiado desde la perspectiva de la capacidad de análisis de los modelos de redes de Petri, de manera que los problemas de compatibilidad se formulan y verifican en términos del modelo de red de Petri correspondiente a la ejecución sincronizada de un *workflow* y un protocolo concretos [20].
- **Workflows científicos.** La plataforma DENEb propone una serie de mecanismos y herramientas que pueden aplicarse fácilmente al ámbito de los *workflows* científicos. A tales efectos, como resultado de una tesis dentro del marco del grupo de investigación se ha desarrollado DVega, una plataforma basada en DENEb para el

modelado de *workflows* científicos flexibles y aplicable a problemas científicos [22].

- Finalmente, la utilización de los modelos de Redes-en-Redes permite el **prototipado rápido** y la **simulación** en escenarios evolutivos y dinámicos, permitiendo la formulación de problemas en términos de análisis de prestaciones, parámetros de calidad de servicio, etc., y proponiendo mecanismos de composición dinámica para la obtención de *workflows* científicos adaptables a entornos de altas prestaciones o de *grid computing*, por ejemplo.

7. Conclusiones

En este trabajo se ha presentado la aplicación del enfoque conversacional y del paradigma de las Redes-en-Redes para el desarrollo e implementación de la plataforma DENEb. Esta plataforma facilita el modelado, desarrollo y ejecución directa a través de los modelos de procesos de negocio que implementan los requisitos de flexibilidad y dinamicidad exigidos para el desarrollo de nuevas aplicaciones. DENEb abarca los problemas derivados de las propuestas, arquitecturas y lenguajes de composición existentes para la integración dinámica de procesos, y aporta un entorno que permite la colaboración de procesos heterogéneos mediante componentes de mediación y herramientas de traducción.

La plataforma desarrollada también permite la integración de protocolos horizontales para implementar ciertas propiedades comportamentales (como por ejemplo las propiedades transaccionales [13]), dando soporte para diferentes mecanismos de despliegue (cliente-servidor, punto a punto, etc.).

Los procesos desarrollados en DENEb son capaces de colaborar y cooperar de forma flexible para componer, fácilmente, procesos de valor añadido. Desde el punto de vista de la ejecución de procesos, DENEb extiende el concepto de adquisición dinámica no sólo a los componentes que forman el sistema, sino también a los aspectos relacionados con la lógica interna de los procesos. Éstos pueden reconfigurar su lógica de negocio y de interacción en tiempo de ejecución, y seleccionar e interaccionar con nuevos *partners* compartiendo, cuando sea necesario, protocolos

no consensuados previamente. Igualmente, los *workflows* pueden ser reconfigurados dinámicamente conforme a los requisitos de negocio y a las estrategias organizativas. Además, aunque las *Reference nets* se han utilizado para los aspectos relativos a la implementación tanto de *workflows* como de protocolos, esta decisión es compatible con los estándares basados en tecnologías de servicios Web.

Las características de adquisición y carga dinámica de las que goza la plataforma abren nuevos retos en forma de nuevas líneas de investigación abiertas y que han dado lugar a varias tesis en el grupo de trabajo en el que se ha desarrollado DENEb. Concretamente, estas líneas abarcan la composición dinámica, la integración de estándares para el modelado de procesos, cuestiones de compatibilidad semántica entre procesos, computación científica basada en *grid*, prototipado rápido y simulación.

Agradecimientos

Este trabajo se ha llevado a cabo en el marco del proyecto TIN 2010-17905, financiado por el Ministerio de Ciencia y Tecnología de España.

Referencias

- [1] Papazoglou, M.P., Traverso, P., Dustdar, S., Leymann, F.: Service-oriented computing: State of the art and research challenges. *Computer*, vol. 40, 2007, pp. 38–45.
- [2] Commission of the European Communities: 3S Green Paper on Software and Service Architectures, Infrastructures and Engineering. Technical report, 2007.
- [3] Mecella, M., Pernici, B., Craca, P.: Compatibility of e-Services in a Cooperative Multi-platform Environment. In: Second International Workshop on Technologies for E-Services – TES 2001, Springer-Verlag, 2001, pp. 44–57.
- [4] Schmidt, R.: Web services based architectures to support dynamic interorganizational business processes. In: International Conference on Web Services - Europe 2003 – ICWS-Europe'03. Volume 2853 of LNCS., Springer, 2003, pp. 123–136.

- [5] Hanson, J.E., Nandi, P., Levine, D.W.: Conversation-enabled web services for agents and e-business. In: International Conference on Internet Computing, 2002, pp. 791–796.
- [6] Ardissono, L., Petrone, G., Segnan, M.: Enabling flexible interaction with web services. In: Extending Web Service Technologies. The use of Multi-Agent approaches. Springer Verlag, 2004, pp. 187–208.
- [7] BPEL 2.0 specification - OASIS. (2007). [Online]. Available: <http://docs.oasis-open.org/wsbpel/>
- [8] Murata T. Petri nets: Properties, analysis and applications. Proceedings of IEEE, vol. 77, 1989, pp. 541–580.
- [9] van der Aalst, W., Hee, K. Workflow Management: Models, Methods, and Systems. MIT Press: Cambridge, MA, USA, 2004.
- [10] Girault, C., Valk, R. Petri Nets for Systems Engineering – A Guide to Modeling, Verification, and Applications. Springer, 2003.
- [11] Kummer O. Introduction to Petri nets and Reference nets. Sozionik Aktuell 2001; 1:1–9. ISSN 1617-2477.
- [12] Kummer, O., Wienberg, F. Renew - the Reference net workshop. Tool Demonstrations, 21st International Conference on Application and Theory of Petri Nets, 2000, pp. 87–89.
- [13] Fabra, J., Álvarez, P., Bañares, J.A., Ezpeleta, J. A framework for the development and execution of horizontal protocols in open BPM systems. 4th Int. Conf. on Business Process Management – BPM 2006, pp. 209–224.
- [14] Fabra, J., Álvarez, P., Bañares, J.A., Ezpeleta, J. Runtime Protocol Binding: Flexible Service Integration By Means Of Flexible Service Interactions. 2008 Int. Conf. on Services Computing – SCC 2008, IEEE Computer Society Press, 2008, pp. 291–298.
- [15] Fabra, J., Álvarez, P., Ezpeleta, J. DRLinda: A Distributed Message Broker For Collaborative Interactions Among Business Processes. 8th Int. Conf. on Electronic Commerce and Web Technologies – EC-Web'07, Springer Verlag, 2007, pp. 212–221.
- [16] Fabra, J., Álvarez, P. BPEL2DENEb: Translation of BPEL Processes to Executable High-level Petri Nets. Fifth International Conference on Internet and Web Applications and Services - ICIW 2010, IEEE Computer Society, 2010, pp. 496-505.
- [17] Fabra, J., Peña, J. Ruiz-Cortés, A., Ezpeleta, J. Enabling the evolution of service-oriented solutions using an UML2 profile and a Reference Petri nets execution platform. Third International Conference on Internet and Web Applications and Services - ICIW 2008, pp. 8-13.
- [18] de Castro, V., Fabra, J., Álvarez, P., Marcos, E. Integración de SOD-M y DENEb: un marco para la ejecución de modelos de procesos de negocio. VI Jornadas Científico-Técnicas en Servicios Web y SOA - JSWEB 2010.
- [19] Fabra, J., Álvarez, P., Ezpeleta, J. Development of Configurable E-Marketplaces Based On a Flexible Management Of E-Negotiation Protocols. 2008 International Conference on E-Commerce, 2008, pp. 133–200.
- [20] Ibáñez, M.J., Álvarez, P., Ezpeleta, J. Checking Necessary Conditions for Control and Data Flow Compatibility between Business and Interaction Logics in Web Processes. 6th IEEE European Conference on Web Services, 2008, pp. 92–101.
- [21] Fabra, J., Álvarez, P., Bañares, J.A., Ezpeleta, J. Integración en DENEb de componentes para la conectividad dinámica de los procesos Web. Aplicación a escenarios de gestión de emergencias basados en Sensor Web. Novática, vol. 197, pp. 47-52, 2009.
- [22] Tolosana-Calasanz, R., Bañares, J.A., Álvarez, P., Ezpeleta, J. On Interlinking of Grids: A Proposal for Improving the Flexibility of Grid Service Interactions. Third International Conference on Internet and Web Applications and Services - ICIW 2008, pp. 714-720.