



# XII JORNADAS DE CIENCIA E INGENIERÍA DE SERVICIOS (JCIS2016)

---

*Carlos Canal (eds.)*



Ediciones Universidad  
**Salamanca**

JCIS



## XII Jornadas de Ciencia e Ingeniería de Servicios (JCIS2016)



CARLOS CANAL (Ed.)

# XII Jornadas de Ciencia e Ingeniería de Servicios (JCIS2016)

  
Ediciones Universidad  
**Salamanca**

# AQUILAFUENTE, 220



Ediciones Universidad de Salamanca y  
de cada autor

Motivo de cubierta:  
Diseñadora María Alonso Miguel

1.º edición: septiembre, 2016

ISBN: 978-84-9012-628-8 (PDF)

Ediciones Universidad de Salamanca  
[www.eusal.es](http://www.eusal.es)  
[eusal@usal.es](mailto:eusal@usal.es)

*Realizado en España – Made in Spain*


*Todos los derechos reservados.*


*Ni la totalidad ni parte de este libro pueden reproducirse ni transmitirse sin permiso escrito de Ediciones Universidad de Salamanca*


Obra sometida a proceso de  
evaluación mediante sistema de revisión por pares a ciegas a tenor de las normas del congreso



Usted es libre de: Compartir — copiar y redistribuir el material en cualquier medio o formato  
Ediciones Universidad de Salamanca no revocará mientras cumpla con los términos:

 Reconocimiento — Debe reconocer adecuadamente la autoría, proporcionar un enlace a la licencia e indicar si se han realizado cambios. Puede hacerlo de cualquier manera razonable, pero no de una manera que sugiera que tiene el apoyo del licenciador o lo recibe por el uso que hace.

 NoComercial — No puede utilizar el material para una finalidad comercial.

 SinObraDerivada — Si remezcla, transforma o crea a partir del material, no puede difundir el material modificado.



Ediciones Universidad de Salamanca es miembro de la UNE  
Unión de Editoriales Españolas [www.une.es](http://www.une.es)



Catalogación de editor en ONIX accesible en <https://www.dilve.es/>

## Prólogo

Los servicios software se han convertido en un factor clave en el crecimiento de cualquier economía desarrollada. Este hecho ha motivado el interés de los distintos actores económicos por desarrollar una Ciencia de los Servicios, también conocida desde una perspectiva más amplia como Ciencia, Gestión e Ingeniería de los Servicios (SSME).

En este marco, las XII Jornadas de Ciencia e Ingeniería de Servicios (JCIS 2016) se constituyen como un foro de discusión e intercambio de conocimiento y experiencias. El interés no sólo se centra en los nuevos avances científicos, sino también en las tecnologías existentes en torno a la computación orientada a servicios y los procesos de negocio, las nuevas prácticas de ingeniería de servicios y las lecciones aprendidas por medio de experiencias reales. Desde sus inicios, JCIS propicia el encuentro de todas las comunidades relacionadas con el ámbito de la Ingeniería de Servicios incluyendo Servicios Web, SOA, Procesos de Negocio, etc. Con este enfoque pretendemos mantener e incrementar la comunicación y el intercambio de experiencias entre los profesionales de la industria, la administración y los investigadores más relevantes en las áreas involucradas.

JCIS 2016 es una conferencia organizada bajo los auspicios de Sistedes (Sociedad de Ingeniería del Software y Tecnologías de Desarrollo de Software) y celebrada en Salamanca dentro del V Congreso Español de Informática (CEDI 2016).





## Comité directivo

Pedro Álvarez (Universidad de Zaragoza)  
Jesús Arias (Universidad Carlos III de Madrid)  
Enrique Beltrán (Software AG)  
Pere Botella (Universitat Politècnica de Catalunya)  
Valeria de Castro (Universidad Rey Juan Carlos)  
Félix García (Universidad de Castilla-La Mancha)  
Manuel Lama (Universidad de Santiago de Compostela)  
Ozelin López (pi Ltd.)  
Esperanza Marcos (Universidad Rey Juan Carlos)  
Juan Manuel Murillo (Universidad de Extremadura)  
Joan Pastor (Universitat Oberta de Catalunya)  
Juan Pavón (Universidad Complutense de Madrid)  
Vicente Pelechano (Universitat Politècnica de València)  
Antonio Ruiz Cortés (Universidad de Sevilla)  
Francisco Ruiz (Universidad de Castilla-La Mancha)

## Comité de programa

Carlos Canal (Presidente, Universidad de Málaga)  
Silvia Acuña (Universidad Autónoma de Madrid)  
Martín Álvarez (Fundación CTIC y W3C Spain Office)  
Pedro Álvarez (Universidad de Zaragoza)  
Jesús Arias (Universidad Carlos III de Madrid)  
Leire Bastida (TECNALIA Research & Innovation)  
Jesús Bermejo (Schneider Electric)  
Javier Berrocal (Universidad de Extremadura)  
Carlos Bobed (Universidad de Zaragoza)  
Juan Boubeta-Puig (Universidad de Cádiz)  
Hilario Canós (Universitat Politècnica de València)  
Josep Carmona (Universitat Politècnica de Catalunya)  
Rafael Corchuelo (Universidad de Sevilla)  
Javier Cubo (Universidad de Málaga)  
Valeria de Castro (Universidad Rey Juan Carlos)  
Andrea Delgado (Universidad de la Republica - Uruguay)  
Javier Fabra (Universidad de Zaragoza)  
Félix García (Universidad de Castilla-La Mancha)  
Fernando González-Ladrón de Guevara (Universitat Politècnica de València)  
Juan Hernández (Universidad de Extremadura)  
Jon Iturrioz (Universidad del País Vasco)

Emilio Labra (Universidad de Oviedo)  
Manuel Lama (Universidad de Santiago de Compostela)  
Diego López (Telefonica I+D)  
Diego López de Ipiña (Universidad de Deusto)  
Marcos López Sanz (Universidad Rey Juan Carlos)  
Jordi Marco (Universitat Politècnica de Catalunya)  
Juan Manuel Murillo (Universidad de Extremadura)  
Marc Oriol (Universitat Politècnica de Catalunya)  
Guadalupe Ortiz (Universidad de Cádiz)  
Joan Pastor (U. Oberta de Catalunya y U. Politècnica de Catalunya)  
Marta Patiño (Universidad Politécnica de Madrid)  
Juan Pavón (Universidad Complutense de Madrid)  
Vicente Pelechano (Universitat Politècnica de València)  
María Penadés (Universitat Politècnica de València)  
Manuel Resinas (Universidad de Sevilla)  
María Ribera (Universitat Politècnica de Catalunya)  
Raúl Romero (Universidad de Córdoba)  
Mercedes Ruiz (Universidad de Cádiz)  
Francisco Ruiz (Universidad de Castilla-La Mancha)  
Antonio Ruiz Cortés (Universidad de Sevilla)  
Victoria Torres (Universitat Politècnica de València)  
Javier Troya (Universidad de Sevilla)  
Antonio Vallecillo (Universidad de Málaga)

# ÍNDICE

## Descubrimiento, predicción y verificación de procesos de negocio

A Recommender System for Process Discovery JOEL RIBEIRO, JOSEP CARMONA, MUSTAFA MISIR, MICHELE SEBAG.....	17
Servitización: Un estudio de Técnicas de Modelado de Negocio VALERIA DE CASTRO, ESPERANZA MARCOS, JUAN M. VARA, ELOÍSA DÍAZ, MARÍA LUZ MARTÍN.....	19
Towards a general architecture for predictive monitoring of business processes ALFONSO E. MÁRQUEZ-CHAMORRO, MANUEL RESINAS, ANTONIO RUIZ-CORTÉS.....	29
Verification, Validation of UML Artifact-centric Business Process Models MONTSERRAT ESTAÑOL, MARIA-RIBERA SANCHO, ERNEST TENIENTE.....	33
Automating Correctness Verification of Artifact-Centric Business Process Models DIANA BORREGO, RAFAEL M. GASCA, MARÍA TERESA GÓMEZ-LÓPEZ.....	35
Process Mining: Past, Present, (Likely) Future JOSEP CARMONA.....	37

## Ecosistemas de servicios, personas y cosas

Reconfiguration of Service Failures in DAMASCo using Dynamic Software Product Lines JAVIER CUBO, NADIA GÁMEZ, ERNESTO PIMENTEL, LIDIA FUENTES.....	45
Automatización de la etapa de análisis para la aplicación de la técnica de prueba metamórfica a composiciones de servicios WS-BPEL M. CARMEN DE CASTRO-CABRERA, KEVIN J. VALLE-GÓMEZ, INMACULADA MEDINA-BULO.....	49
Opening up Context-Aware Service Compositions to End-Users IGNACIO MANSANET, VICTORIA TORRES, PEDRO VALDERAS, JAVIER BERROCAL, JOSÉ GARCÍA-ALONSO, JUAN M. MURILLO, CARLOS CANAL.....	55
Construyendo Perfiles Virtuales Mediante el Procesamiento de Eventos Complejos JAVIER BERROCAL, JUAN BOUBETA-PUIG, CARLOS CANAL, JOSÉ GARCÍA-ALONSO, JUAN M. MURILLO, GUADALUPE ORTIZ.....	59
La Web de las Cosas se nos viene encima CARLOS CANAL, JUAN M. MURILLO.....	69

## Los servicios y la Nube

<b>Bidimensional Cross-Cloud Management with Brooklyn, TOSCA</b> JOSÉ CARRASCO, JAVIER CUBO, FRANCISCO DURÁN, ERNESTO PIMENTEL.....	75
<b>Reconfigurando Aplicaciones Multi-Cloud con Líneas de Producto Software Dinámicas</b> JAVIER CUBO, NADIA GÁMEZ, ERNESTO PIMENTEL, LIDIA FUENTES.....	79
<b>Una aproximación DevOps para el Desarrollo Dirigido por Modelos de Servicios Cloud</b> JULIO SANDBALÍN-GUZMÁN, MIGUEL ZÚÑIGA PRIETO, EMILIO INSFRAN, SILVIA ABRAHÃO, CARLOS CANO.....	83
<b>Automatic Generation of Purchasing Plans for Cloud Services</b> OCTAVIO MARTÍN-DÍAZ, JOSÉ MARÍA GARCÍA, PABLO FERNÁNDEZ, ANTONIO RUIZ-CORTÉS.....	93
<b>Governing the service-chain: Challenges ahead</b> PABLO FERNÁNDEZ.....	103

## Elasticidad y variabilidad en procesos de negocio

<b>VIVACE: A framework for the systematic evaluation of variability support in process-aware information systems</b> CLARA AYORA, VICTORIA TORRES, BARBARA WEBER, MANFRED REICHERT, VICENTE PELECHANO.....	109
<b>Reparación de alignments en modelos de proceso similares</b> BORJA VÁZQUEZ-BARREIROS, SEBASTIAAN VAN ZELST, JOOS BUIJS, MANUEL LAMA, MANUEL MUCIENTES.....	111
<b>Defining PPIs for Process Variants based on Change Patterns</b> BEDILIA ESTRADA-TORRES, VICTORIA TORRES, ADELA DEL RÍO-ORTEGA, MANUEL RESINAS, VICENTE PELECHANO, ANTONIO RUIZ-CORTÉS.....	121
<b>Incertidumbre de Datos en el Modelado de Procesos de Negocio</b> PRISCILL ORUE, CARMEN MORCILLO, ANTONIO VALLECILLO.....	125
<b>Programming Elasticity, Commitment in Dynamic Processes</b> PABLO FERNÁNDEZ, HONG-LINH TRUONG, SCHAHRAM DUSTDAR, ANTONIO RUIZ CORTÉS.....	129

## Servicios Web

<b>Monitoring the service-based system lifecycle with SALMon</b> MARC ORIOL, XAVIER FRANCH, JORDI MARCO.....	133
<b>Modelling Service Level Agreements for Business Process Outsourcing Services</b> ADELA DEL RÍO-ORTEGA, ANTONIO MANUEL GUTIÉRREZ, AMADOR DURÁN, MANUEL RESINAS, ANTONIO RUIZ-CORTÉS.....	135
<b>Supporting Compensations with WS-Agreement</b> CARLOS MÜLLER, PABLO FERNÁNDEZ, OCTAVIO MARTÍN-DÍAZ, ANTONIO MANUEL GUTIÉRREZ, MANUEL RESINAS, ANTONIO RUIZ-CORTÉS.....	137
<b>MigraSOA: Migrando aplicaciones web legadas hacia arquitecturas orientadas a servicios (SOA) Largo</b> ENCARNA SOSA-SÁNCHEZ, PEDRO J. CLEMENTE, ÁLVARO PRIETO, JOSÉ M. CONEJERO, ROBERTO RODRÍGUEZ-ECHEVERRÍA.....	147

SOA 2.0 para el Control y Alerta de Riesgos para la Salud Derivados  
de la Calidad del Aire

ALFONSO GARCÍA DE PRADO, GUADALUPE ORTIZ, JUAN BOUBETA-PUIG, ADOLFOR. DE SOTO.... 157

**El factor humano en los procesos de negocio**

**Recompilación de procesos de educación a partir de registros de eventos**

JUAN C. VIDAL, BORJA VÁZQUEZ-BARREIROS, MANUEL LAMA, MANUEL MUCIENTES..... 169

**BPMS-Game: Herramienta para la Gamificación de Procesos de Negocio**

JAVIER MANCEBO, FÉLIX GARCÍA..... 171

**Análisis inteligente de flujos de trabajo sociales**

MANUEL LAMA, PEDRO ÁLVAREZ, MANUEL OCAÑA, MANUEL MUCIENTES, JOAQUÍN EZPELETA,  
MIGUEL ÁNGEL GARRIDO, ALBERTO BUGARÍN..... 181

**RALph: A Graphical Notation for Resource Assignments in Business Processes**

CRISTINA CABANILLAS, DAVID KNUPLESCH, MANUEL RESINAS, MANFRED REICHERT,  
JAN MENDLING, ANTONIO RUIZ-CORTÉS..... 191



# Descubrimiento, predicción y verificación de procesos de negocio





# A Recommender System for Process Discovery\*

Joel Ribeiro<sup>1</sup>, Josep Carmona<sup>1</sup>, Mustafa Misir<sup>2</sup>, and Michele Sebag<sup>2</sup>

<sup>1</sup> Universitat Politècnica de Catalunya, Spain.  
{jrbeiro, jcarmona}@lsi.upc.edu

<sup>2</sup> TAO, INRIA Saclay - CNRS - LRI, Université Paris Sud XI, Orsay, France.  
{mustafa.misir,michele.sebag}@lri.fr

## Summary of the contribution

The ability of monitoring process executions within information systems yields large-scale event log files. These files can be processed using the so-called *process mining* approaches, at the crossroad of *business intelligence* and *data mining* techniques. Process mining is positioning as the perfect candidate to support information systems in the *big data* era.

Process mining is defined as the extraction of valuable information from event logs, aimed at strategic insight into the business processes [2]. Process mining mainly includes *process discovery*, *conformance checking* and *enhancement*. Discovery techniques aim at the behavioral modeling of the business process underlying the event logs. Conformance techniques check the compatibility of a process model with regard to a set of event logs. Enhancement techniques enrich a process model based on additional process information available in the event log.

Over the last decade, several algorithms for process discovery and process conformance have been proposed. Still, it is well-accepted that there is no dominant algorithm in any of these two disciplines, and then it is often difficult to apply them successfully. Most of these algorithms need a close-to expert knowledge in order to be applied satisfactorily. In this paper, we present a recommender system that uses portfolio-based algorithm selection strategies to face the following problems: to find the best discovery algorithm for the data at hand, and to allow bridging the gap between general users and process mining algorithms.

In detail, the contribution of the paper is an integrated process discovery framework achieving algorithm selection based on machine learning techniques. Formally, this framework elaborates on the *Algorithm Recommender System* (ARS) approach [1], based on using a dataset that reports the results of some algorithms in the portfolio on a set of problem instances; its generality is witnessed as it has been applied successfully in domains such as *Constraint Satisfaction* and *Optimization*.

ARS is integrated within a framework to evaluate process discovery algorithms [3]. We have developed a server-client architecture along the training-test

\* This work has been partially supported by funds from the Spanish Ministry for Economy and Competitiveness (MINECO), the European Union (FEDER funds) under grant COMMAS (ref. TIN2013-46181-C2-1-R).

principle used in machine learning. The server achieves lifelong learning, continuously running process discovery experiments to enrich its database reporting the performances of algorithms on case studies (event logs). This database is exploited using ARS, continuously increasing the system knowledge. This knowledge is then disseminated to the clients, that use it to predict the best algorithm on their current event log. The client is implemented as a ProM [4] plugin.

## Information about the Publication

**Citation:** Joel Ribeiro, Josep Carmona, Mustafa Misir, Michele Sebag: A Recommender System for Process Discovery. 12th International Conference on Business Process Management, BPM 2014, Haifa, Israel, September 7-11, 67-83. Lecture Notes in Computer Science 8659, Springer 2014, ISBN 978-3-319-10171-2

## References

1. M. Misir and M. Sebag. Algorithm Selection as a Collaborative Filtering Problem. Technical report, INRIA, 2013.
2. W.M.P. van der Aalst. *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer, Berlin, 2011.
3. S. vanden Broucke, C. Delvaux, J. Freitas, T. Rogova, J. Vanthienen, and B. Baensens. Uncovering the Relationship between Event Log Characteristics and Process Discovery Techniques. In *Proceedings of the 9th Workshop on Business Process Intelligence (BPI 2013)*, 2013.
4. H.M.W. Verbeek, J.C.A.M. Buijs, B.F. van Dongen, and W.M.P. van der Aalst. ProM 6: The Process Mining Toolkit. In *Demo at the 8th International Conference on Business Process Management*, volume 615 of *CEUR-WS*, pages 34–39. 2010.

# Servitización: Un estudio de Técnicas de Modelado de Negocio

Valeria De Castro, Esperanza Marcos, Juan M. Vara, Eloísa Díaz, María Luz Martín

Grupo de Investigación Kybele, Universidad Rey Juan Carlos

Calle Tulipán S/N, 28933 Móstoles, Madrid, España  
{valeria.decastro, esperanza.marcos, juanmanuel.vara, eloisa.diaz@urjc.es, luz.martin@urjc.es}@urjc.es

**Resumen.** El concepto de servitización se puede resumir en la idea de ofrecer servicios en combinación con productos con el objetivo de mejorar la oferta de valor al cliente. Para afrontar exitosamente un proceso de servitización, las empresas necesitan rediseñar su modelo de negocio. Definir el modelo de negocio implica un proceso de innovación y la conceptualización de la idea de negocio de una organización. El objetivo de este trabajo es analizar brevemente el proceso de servitización en la literatura y presentar un estudio sobre las técnicas y metodologías para el modelado de negocio que pueden ayudar en este proceso.

**Keywords:** Servitización, Servicios, Modelo de Negocio.

## 1 Introducción

La Servitización puede ser entendida como el proceso de aumentar el valor añadiendo servicios a los productos [9]. El proceso de servitización puede verse como el desarrollo de las capacidades de innovación de una organización, en el sentido de que en lugar de ofrecer sólo productos se pasa a ofrecer sistemas de producto-servicio [5,12].

Muchas empresas afrontan en la actualidad procesos de servitización. Así, pueden verse ejemplos como los del sector de la Automoción, en los que el foco principal de las compañías ya no es sólo el producto (el coche), sino todos los servicios, cada vez más complejos y orientados al cliente, que se ofrecen a su alrededor y que pretenden mejorar su experiencia a la hora de conducir.

A la hora de afrontar el proceso de servitización con éxito, la empresa necesita rediseñar su modelo de negocio [1]. El cambio hacia una orientación de servicios en las empresas es deseable, sin embargo, no está exento de dificultades, debido a la necesaria transformación del modelo de negocio. El análisis y la construcción del modelo de negocio constituyen así la parte primaria de un proceso de servitización.

El modelo de negocio describe las bases sobre las cuales una empresa crea, proporciona y capta valor [6]. La definición de un modelo de negocio implica identificar la manera en que la empresa aporta valor a los clientes, atrae a los clientes a pagar por dicho valor y convierte esos pagos a beneficio [9].

En la actualidad existen diversas técnicas y métodos para el modelado de negocio que permiten explorar una idea de negocio con más o menos detalles, a la vez que ayudan a entender, conceptualizar y representar aquellos servicios que añaden valor a una organización. Entre ellos modelos como el *Canvas Business Model* [6], el *Value Model* [3], y modelos más orientados a procesos como el diagrama *Process Chain Network* (PCN) [8], el *Service Blueprint* [2] o el modelo *Business Process Model and Notation* (BPMN<sup>1</sup>).

Tales técnicas o métodos tienen aspectos que los diferencian, pero son, en muchos casos, complementarios. El objetivo de este trabajo es presentar brevemente algunas ventajas y desventajas que ofrecen cada uno de ellos a la hora de afrontar un proceso de servitización. Por razones de espacio en este trabajo nos centraremos en el análisis de tres modelos de negocio que son particularmente diferentes y útiles a la hora de modelar y analizar una propuesta de valor que puede ser un servicio o una oferta combinada de producto-servicio. Analizaremos en detalle el Modelo de Valor (Value Model), Modelo Canvas y Diagrama PCN. Más adelante referiremos también brevemente algunas características de los modelos Service Blueprint y BPMN.

En este trabajo se presenta en primer lugar una breve descripción sobre el proceso de servitización y se describen posteriormente algunas de las técnicas y metodologías de modelado de negocio que pueden ser especialmente útiles en dicho proceso. Para finalizar, se recogen las conclusiones y se analizan algunas de las principales líneas de trabajo abiertas.

## 2 El proceso de Servitización

Un servicio puede ser definido como aquella actividad económica que crea valor y proporciona beneficio a los clientes. Un servicio es un acto o desempeño que ofrece una parte a otra. El proceso de prestación del servicio puede estar vinculado a un producto físico, pero el desempeño es en esencia intangible y no da como resultado la propiedad de los factores de producción [4]. El término servitización fue acuñado por Vandermerwe y Rada [10], quienes la definieron como “la oferta incrementada de paquetes más completos al mercado”, es decir, de combinaciones de bienes, servicios, apoyo, auto-servicio y conocimiento, con el fin de añadir valor a la oferta de productos básicos.

Desde que dichos autores analizaran por vez primera el proceso de servitización, en la literatura han surgido numerosos trabajos que ofrecen definiciones diferentes, tales como: “añadir componentes de servicio extra a los productos esenciales” [11], “un paquete integrado de bienes y servicios” [7], y otras más centradas en la servitización como un proceso de innovación: que definen a la servitización como “la innovación en las habilidades y procesos de una organización de forma que se pueda crear una mejor

---

<sup>1</sup> <http://www.bpmn.org/>

relación cliente-organización a través de un cambio de la venta de productos a la venta de sistemas producto-servicio” [5], o como “una innovación en el modelo de negocio en la que los fabricantes de productos expanden el alcance de sus transacciones con los clientes a través del ofrecimiento de productos relacionados con servicios” [12].

En base a estas y otras aportaciones previas de la literatura, es factible definir el proceso de servitización como un “proceso de innovación en el modelo de negocio de una organización, que conduce a la oferta integrada de bienes y servicios, mejorando la satisfacción de las necesidades de los clientes y permitiendo la generación de ventajas competitivas”.

Para implantar una estrategia de servitización de forma exitosa, las empresas necesitan cambiar sus estrategias, operaciones y su cadena de valor entre otras cosas; en resumen, necesitan introducir cambios en el modelo de negocio.

El modelado de negocio constituye una etapa clave del proceso de servitización en la que se busca conceptualizar la idea de negocio de una organización. Dicho modelo debe ayudar a entender, analizar y representar aquellos nuevos servicios que pueden añadir valor a los productos que se ofrecen. En la siguiente sección se presentarán tres técnicas y métodos de modelado de negocio que, acorde con sus características, pueden ser especialmente relevantes a la hora de llevar adelante un proceso de servitización.

### 3 Técnicas de Modelado de Negocio

El modelo de valor o *Value Model* [3] es un modelo de negocio que permite la representación una idea de negocio gráficamente como un conjunto de intercambios de valor y actividades de valor entre distintos actores de un negocio. El modelo de valor modela actores (unitarios o segmentos de mercado) que intercambian entre ellos objetos de valor (bienes, dinero o servicios). En el modelo se representan también actividades de valor, que son las actividades que realiza un actor para la obtención de un beneficio. Una actividad de valor puede ser, por ejemplo, un servicio que ofrece un actor.

En un proceso de servitización el modelo de valor puede ser útil a la hora de combinar actividades de valor que producen un bien (ej. fabricar un coche) con actividades de valor que ofrecen servicios (ej. servicio post-venta). El modelo ofrece características que permite analizar y evaluar el beneficio económico que se podría obtener a través de la venta de productos, de los servicios y de la oferta integrada de ambos a la vez.

En la Figura 1 se representa un modelo de valor con una vista simplificada de lo que puede ser un servicio de alquiler de bicicletas como es el caso de BiciMad<sup>2</sup>. Como puede verse, el modelo permite visualizar tanto la actividad de manufactura de la bicicleta (azul – relacionado con la actividad *Bike manufacturing*) como el servicio de alquiler de bicicletas por parte de la organización proveedora (rojo – relacionado principalmente con la actividad *Bike renting*). Dicho modelo muestra, gráficamente, que el servicio de alquiler de bicicletas implica tanto la producción y como el mantenimiento del bien.

---

<sup>2</sup> <http://www.bicimad.com/>

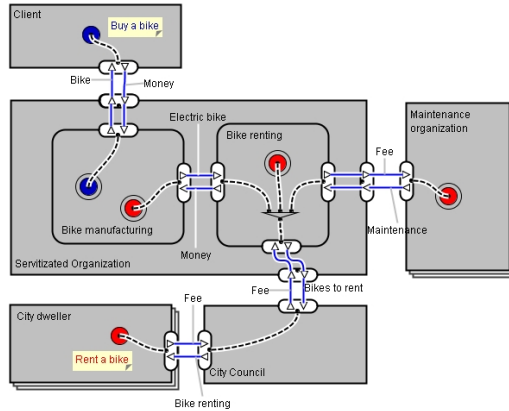


Fig. 1. Modelo de Valor caso BiciMad.

Otro modelo de negocio, y quizás el más utilizado en el ámbito de las empresas, es el **Modelo Canvas** [6] que define un conjunto de conceptos y un tablero que permite describir, visualizar, evaluar y cambiar un modelo un negocio. El modelo de negocio se construye en torno a nueve módulos básicos que representan la forma en la que una empresa consigue obtener beneficios a partir de una propuesta de valor. Tales módulos son: la propuesta de valor (PV); las relaciones con los clientes (RCI), los segmentos de mercado (SM), los canales de distribución (C) y las fuentes de ingresos (FI) que se obtienen de analizar la relación entre la empresa y los clientes (parte derecha del Canvas); y la estructura de costes (EC), las asociaciones claves (AsC), recursos claves (RC) y las actividades claves (AC) que se obtienen de un análisis interno de la empresa y su relación con proveedores o socios de negocio (parte izquierda del Canvas). La Figura 2 muestra un ejemplo de un Modelo Canvas para un servicio de transporte en ciudad a través del cual los clientes pueden alquilar y utilizar un coche en cualquier momento.

El Modelo Canvas puede ayudar a visualizar y evaluar una propuesta de valor en la que se combine un producto y un servicio. La figura 3 representa el Modelo Canvas para el caso de BiciMad. Como puede verse en el tablero Canvas, una desventaja importante de este modelo respecto del anterior, es que no es posible visualizar en el mismo modelo todos los elementos relacionados con distintas propuestas de valor (bienes o servicios) que ofrece la empresa, ya que el modelo, por sus características, sólo permite analizar una propuesta de valor a la vez. El poder ver todas las actividades de “producción” de una empresa a la vez (de bienes o de servicios) puede ser especialmente útil cuando se diseñan ofertas combinadas de productos-servicios.

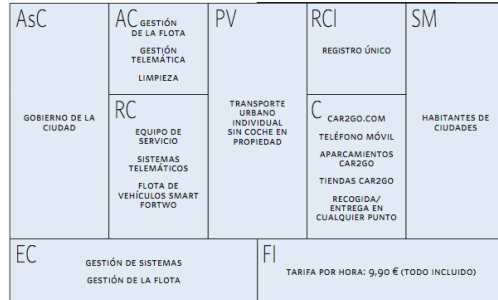


Fig. 2. Ejemplo de Modelo Canvas. Extraído de [6].

La distribución visual en el tablero Canvas de elementos que son claves en el negocio es una de las principales fortalezas de este modelo. En él se incluyen elementos que están presentes también en el modelo de valor como segmento de mercado, clientes, actividades claves, *partners* (asociados o colaboradores) o costes/ingresos; pero aparecen otros, como los recursos claves, los canales de distribución o las formas de establecer relaciones con los clientes que constituyen aspectos importantes en el modelo de negocio de una organización.

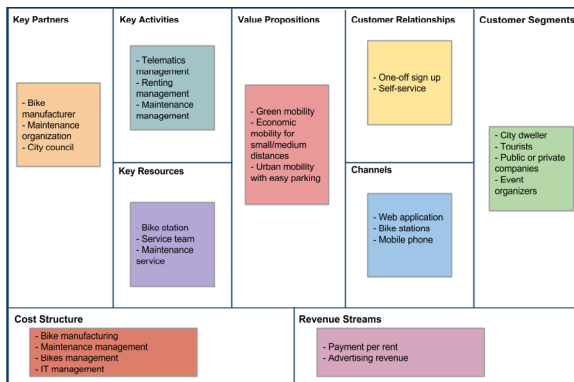


Fig. 3. Modelo Canvas caso BiciMad.

Por último, presentamos el diagrama PCN (*Process-Chain-Network*) que permite la representación y el análisis de servicios como un tipo de proceso [8]. A través de dicho modelo es posible identificar y analizar una propuesta de valor, representar elementos

tangibles en la entrega de un servicio y las características operativas del proceso de entrega del mismo. Los diagramas PCN ofrecen una perspectiva equilibrada y centrada tanto en el cliente y como en el proveedor, enfatizando en las relaciones entre éstos a lo largo del proceso.

La Figura 4 muestra un ejemplo de un diagrama PCN para una pizzería. Un diagrama PCN representa un proceso que es realizado por una entidad o entidades que actúan sobre recursos (físicos, conocimiento, energía, etc.). Un proceso tiene siempre un propósito (en el caso del ejemplo, servir pizzas) que busca mejorar la entidad o entidades involucradas en el proceso, es decir crear valor. En el modelo se representan: interacciones directas (*direct interaction*) con otras entidades tales como clientes o proveedores; interacciones subrogadas (*subrogate interaction*) lo que significa que la entidad que está llevando a cabo el proceso utiliza recursos no-humanos de otras entidades; y, actividades de procesamiento independiente (*independent processing*) lo que significa que el proceso no involucra interacciones directas ni subrogadas con otras entidades. El triángulo en la parte superior del diagrama representa de forma simbólica el grado de control que se tiene sobre el proceso.

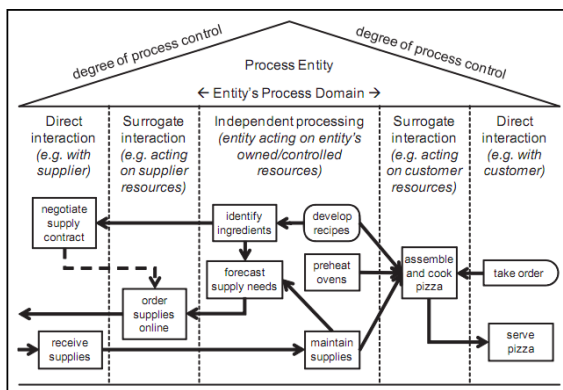


Fig. 4. Ejemplo de Diagrama PCN. Extraído de [8].

El diagrama PCN, de igual manera que el resto de los modelos presentados, permite realizar diferentes tipos de análisis a la hora de diseñar o innovar el modelo de negocio en procesos de servitización. Este modelo difiere de los anteriores en que representa una vista del proceso de entrega de un servicio. Dicho proceso podría involucrar tanto actividades de manufactura como actividades de entrega de un servicio.

La Figura 5 muestra una parte del Diagrama PCN para el momento concreto de obtener una bicicleta de BiciMad en la que se representan tanto la entidad cliente como la entidad BiciMad. Puede verse en la figura que el diagrama permite representar el proceso del servicio identificando cada paso en las distintas zonas de interacción de



cada entidad. El diagrama permitiría representar además otras entidades como la proveedora de bicicletas o la empresa de mantenimiento y los procesos de interacción con estas entidades.

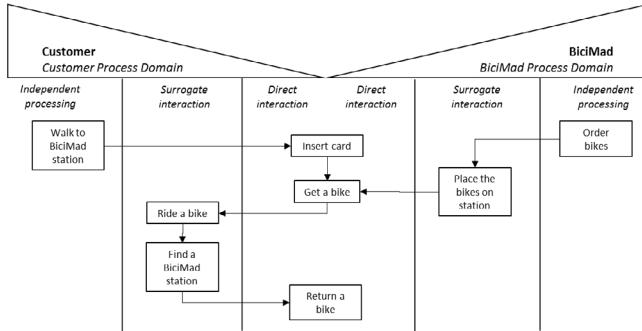


Fig. 5. Diagrama PCN caso BiciMad.

Este modelo presenta además importantes características que permiten analizar e innovar sobre el modelo de servicio. Una vez identificada, por ejemplo, la propuesta de valor y representado el diagrama correspondiente, sería posible analizar y cambiar el modelo de servicio moviendo actividades de una zona de interacción a otra. Así, por ejemplo, sería posible buscar una mejora en la eficiencia del proceso reduciendo las actividades que se representan en zonas de interacción con otras entidades, o bien buscar más una personalización/especialización del servicio moviendo partes de procesamiento independiente a zonas de interacción subrogada o directa con los clientes.

#### 4 Comparativa

La Tabla 1 presenta una comparativa de las tres técnicas analizadas en este trabajo. En ella se resumen los principales elementos de modelado presentes en cada técnica y las posibilidades que ofrecen a la hora de representar una oferta integrada de productos y servicios como sucede en los casos de servitización. En tabla se resumen además otros factores comentados previamente en este trabajo como si representan un proceso, el tipo de técnica y ventajas añadidas del modelo respecto de la posibilidad de realizar diferentes tipos de análisis sobre el modelo de negocio.

Destacando algunas características de otros modelos que no hemos incluido en este análisis, podemos decir que el Service Blueprint es un modelo que presenta también un proceso, pero aportando otra perspectiva diferente y particularmente importante en el ámbito de los servicios que es una visualización clara de qué parte de los servicios sucede en el *back-end* y qué parte en el *front-end*, además de incluir una representación

de las evidencias físicas que no se diferencian explícitamente en otros modelos. El modelo BPMN consideramos que es un modelo de más bajo nivel que puede ser especialmente útil a la hora de alinear el modelo de negocio de una organización con sus procesos de negocio y los servicios desde el punto de vista de las tecnologías de la información que lo soporten.

**Tabla 1.** Comparativa de Modelos de Negocio analizados en el trabajo.

	Modelo de Valor	Modelo Canvas	Diagrama PCN
Elementos presentes en el modelo			
Propuesta de Valor	Si. Pueden verse varias en un modelo	Si	Si. Cada proceso tiene un objetivo que es crear valor
Segmento de Mercado/Cliente	Si	Si	Si
Canales de Distribución	No explícitamente	Si	No explícitamente
Actividades	Si	Si	Si
Recursos	No explícitamente	Si	No explícitamente
Partners	Si	Si	Si
Costes e Ingresos	Si	Si	No explícitamente
¿Representa proceso?	No	No	Si
Tipo de técnica	Visual	Textual. Plantilla	Visual
Servitización. ¿Es posible visualizar diferentes propuestas de valor (oferta de producto, oferta de servicio, oferta combinada) en el mismo modelo?	Si	No. Cada modelo solo representa una propuesta de valor	No. Un proceso representa una propuesta de valor
Otras ventajas	Permite analizar el beneficio económico obtenido de cada propuesta de valor	Es una representación visual y completa de la estrategia de negocio y permite su evaluación	Permite hacer distintos tipos de análisis e innovación del modelo de negocio

## 5 Conclusiones y líneas de trabajo abiertas

Las actividades de servicios son cada vez más importantes en las empresas industriales. Los fabricantes tienen varias razones por las que incluir servicios a su oferta de bienes, como por ejemplo: facilitar las ventas de los bienes, fortalecer las relaciones con el cliente, crear oportunidades de crecimiento en mercados maduros, responder a la demanda, etc. De hecho cada vez más los clientes demandan servicios y soluciones integradas con la compra de los bienes.

Los servicios son mucho más difíciles de imitar que los bienes y por tanto pueden llegar a suponer una fuente de ventaja competitiva para la empresa. Competir en base a servicios permite a los fabricantes aumentar su margen de beneficios. Estas empresas pueden aumentar su rentabilidad añadiendo servicios a dichos productos y diferenciarse de los competidores a través de los servicios que ofrecen.

Acometer un proceso de servitización en las empresas implica necesariamente revisar varios aspectos que pueden resumirse en un cambio en el modelo de negocio de una organización. El modelado de negocio constituye así una etapa clave en este proceso.

Disponer de un amplio abanico de técnicas para el modelado de negocio y conocer sus principales características, ventajas y desventajas, puede redundar en un mayor conocimiento y capacidad de análisis por parte de las organizaciones a la hora de afrontar un proceso de servitización. En este trabajo se han presentado y analizado brevemente tres técnicas que pueden resultar especialmente útiles en dicho proceso.

En la actualidad, y en relación con este trabajo, se está llevando a cabo una revisión sistemática con la que se pretende ampliar y profundizar el estudio de las técnicas de modelado de negocio. Se trabaja también en el desarrollo de una herramienta que permita integrar los diferentes modelos de negocio estudiados, a la vez que explotar otras capacidades, que la herramienta puede facilitar, como la posibilidad analizar relaciones entre los modelos, realizar transformaciones, hacer análisis de gap entre modelos, etc.

**Agradecimientos.** Esta investigación ha sido parcialmente financiada por el Gobierno de la Comunidad de Madrid bajo el proyecto SICOMORO-CM (S2013/ICE-3006), por el proyecto ELASTIC (TIN2014-52938-C2-1-R) y RCIS (TIN2014-53986-REDT) financiado por el Ministerio de Ciencia e Innovación del Gobierno de España y por el Grupo de Excelencia Investigadora en Service Science, Management and Engineering-GES2ME (Ref. Orgánica 30VCP1G115) co-financiado por la Universidad Rey Juan Carlos y el Banco Santander.

## Referencias

1. Baines, T., Lightfoot H.; Benedettini. O. y Kay, J.M., “The Servitization of manufacturing: A review of literature and reflection on future challenges”, *Journal of Manufacturing Technology Management*, Vol. 20, No. 5, pp. 547-567 (2009).
2. Bitner, M., Ostrom, A., & Morgan, F., *Service Blueprinting: A Practical Technique for Service Innovation*, *California Management Review*, Vol. 50, No. 3, Spring 2008, pp. 66-94 (2008).
3. Gordijn, J., Akkermans, J.M., “Value based requirements engineering: exploring innovative e-commerce idea”. *Requirements Engineering Journal*, Vol. 8, No. 2, pp. 114 -134 (2003).
4. Martín-Peña, M.L. y Díaz-Garrido, E., *Fundamentos de Dirección de Operaciones en Empresas de Servicios*. Madrid: ESIC Editorial (2013).
5. Neely, A., “Exploring the financial consequences of Servitization of manufacturing”, *Operations Management Research*, Vol. 1, pp. 103-118 (2008).
6. Osterwalder, A. y Pigneur, Y., *Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers*. John Wiley and Sons (2010).

7. Robinson, T., Clarke-Hill, C.M. y Clarkson, R., "Differentiation through service: a perspective from the commodity chemicals sector", *Service Industries Journal*, Vol. 22, No. 3, pp. 149-66 (2002).
8. Sampson, S., Visualizing Service Operations. *Journal of Service Research*. Vol. 15, No. 2, pp. 182-198 (2012).
9. Teece, D.J., "Business models, business strategy and innovation". *Long Range Planning*, Vol. 43, pp. 172-194 (2010).
10. Vandermerwe, S., Rada, J., "Servitización of Business: Adding value by adding services", *European Management Journal*, Vol. 6, No. 4, pp. 314-324 (1988).
11. Verstrepren, S. and van Den Berg, R., "Servitization in the automotive sector: creating value and competitive advantage through service after sales", *Global Production Management*, Kluwer Publishers, London, pp. 538-45 (1999).
12. Visnjic, I., Van Looy, B., "Servitización: Disentangling the impact of service business model innovation on manufacturing firm performance", *Journal Operations Management*, Vol. 31, pp. 169-180 (2013).

# Towards a general architecture for predictive monitoring of business processes\*

Alfonso E. Márquez-Chamorro, Manuel Resinas and Antonio Ruiz-Cortés

Dpto. Lenguajes y Sistemas Informáticos, University of Seville, Seville, Spain.  
{amarquez6,resinas,aruiz}@us.es

**Abstract.** Process mining allows the extraction of useful information from event logs and historical data of business processes. This information will improve the performance of these processes and is generally obtained after they have finished. Therefore, predictive monitoring of business process running instances is needed, in order to provide proactive and corrective actions to improve the process performance and mitigate the possible risks in real time. This monitoring allows the prediction of evaluation metrics for a runtime process. In this context, this work describes a general methodology for a business process monitoring system for the prediction of process performance indicators and their stages, such as, the processing and encoding of log events, the calculation of aggregated attributes or the application of a data mining algorithm.

**Keywords:** business process, process mining, predictive monitoring, business process indicator prediction.

## 1 Introduction

Predictive monitoring of business processes is one of the main issues in process mining and aims to predict possible quantifiable metrics of a running process instance. These metrics evaluate the performance of a business process in terms of efficiency and effectiveness and can be related to several cases, a complete case or a specific case event. Some examples of these metrics are the remaining execution time of a process, the likelihood of a fault in the system or the abnormal termination of a running instance.

In this context, the well-known Knowledge discovery in databases (KDD) process can be applied. This process aims to find knowledge in datasets combining with the application of data mining techniques. The stages which compose a KDD process are: data extraction, preprocessing of data, transformation and reduction of data, selection and application of the data mining algorithm, interpretation of the results and evaluation and finally, the knowledge deployment.

In this work, we present a general architecture for the prediction of performance indicators for a business process running instance.

---

\* This work has received funding from the European Commission (FEDER), the Spanish and the Andalusian R&D&I programmes (grants TIN2015-70560-R (BELI), COPAS (P12-TIC-1867) and Juan de la Cierva (JCF 2015)).

## 2 Methodology

This section describes a general architecture for the predictive monitoring of business process. Specifically, this is a generic architecture for the prediction of different process performance indicators, such as time indicators, risk indicators, SLA violation indicators or other performance indicators. The event log of a process will constitute the training dataset of a learning classifier. These event traces are then encoded in sequences or feature vectors that can be interpreted by the classifiers. These sequences are composed by the attributes of the different events of historical traces. For each sequence, a class is assigned. This class corresponds to the value of the indicator which we are aimed to predict. Then, the classifier is trained and generates a predictive model. The trace of an ongoing process instance is used as a test example for the classifier in a determined moment of the execution (checkpoint). Finally, the learned predictive model will determine the predicted value for the process performance indicator. Figure 1 represents the experimental procedure and the different cited stages of the process. These stages of the methodology are summarized in the following:

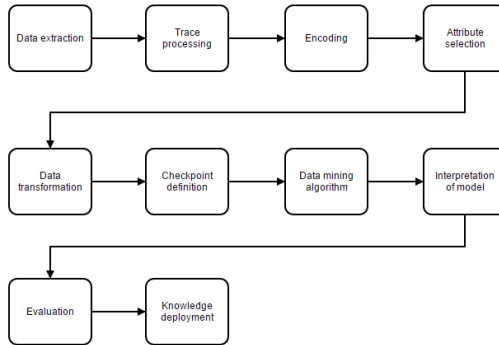


Fig. 1. Experimental procedure scheme.

**Data extraction:** Data are generally provided by information systems that record traces about process executions. Massive amounts of information can be generated by one of these systems which are stored in event logs. Therefore, it is necessary a management of the data for an adequate predictive monitoring process.

**Trace processing:** A classification of process characteristics is described in [1]. These characteristics can be transcribed from the traces of the event logs, according to four different perspectives: the control-flow perspective, related to the order of the activities to be performed in the process, the data-flow perspective, which refers to different attributes attached to the events, the time perspective, referred to various types of duration in the process, such as the duration of an activity or the remaining time of a process, the resource/organization perspective related to the resource that executes a determined event and the conformance perspective which provides answers to several conformance-related questions. During this stage, certain data preprocessing operations can be performed, such as missing values treatment, outliers detection or noise reduction. A resampling of data can be also performed to prevent the unbalanced classes.

**Encoding:** According to the cited perspectives, it is necessary to describe an encoding which stores enough information of the process. Generally, the encoding for a trace includes only the flow perspective. The data-flow perspective is also incorporated in some recent encodings, considering the information data of the events and not only the sequence flow. The encoding usually represents the events and their associated information which compose a process or a part of it.

**Feature selection:** Feature selection is applied to select a subset of the features from the event logs and reduce its dimensions by using a minimal set of features to represent the maximum amount of variance in the data. The aim is to increase the classification accuracy and efficiency of the algorithm by eliminating irrelevant, redundant or correlated attributes. Different evaluation methods and search methods are used for the attribute selection, such as Chi squared test and Recursive feature elimination respectively.

**Data transformation:** Some attributes of the event logs are inconsistent or have to be changed for a better treatment of data. Two typical operations of transformation of data are standardization and normalization. Sometimes, it is necessary to add new features derived from other attributes, *e.g.* elapsed time from the beginning of a process to the current event. These are named aggregated attributes and are calculated in this stage. Other transformations can be performed to extract a higher level information from the log, and can be related to the frequency of the activities, execution times of the events or the discover of patterns in the sequence of activities.

**Checkpoint definition:** Checkpoints indicate points in the execution where a prediction should be carried out. Each checkpoint should be established before an activity in a business process. For each checkpoint, a predictive model has to be generated for the data mining algorithm. This is due to the different training set used at each checkpoint. In addition, predictive models have to be changed over time because the behaviors of users can vary. Selection strategies to define the checkpoints have to be considered [2]. The higher the number of checkpoints, the greater the computational cost of the methodology. On the other hand, a higher number of checkpoints along the process cycle, provides a more accurate model for the predictions. Thus, a trade-off between computational cost and effectiveness of the method must be achieved.

**Data mining algorithm:** According to the process performance metric to be predicted, we have to select one or another data mining algorithm: for predicting a discrete attribute, such as a binary value to determine the fulfilment or violation of a certain constraint [3], we can select a classification method such as decision trees or neural networks. For predicting a continuous attribute, such as the remaining time of completion of a process [4], we can choose a regression method as regression trees or support vector regression. Finally for finding groups of similar items, we can select among the different clustering methods.

**Evaluation:** For the accuracy assessment of predictive monitoring methods, three measures are usually employed. Precision represents the rate of correctly predicted process instances. Recall reflects the proportion of predicted process instances divided by the total number of instances. Root-mean squared error (RMSE) calculates the error between the real values and the predicted values.

**Interpretation of model:** Generally, the predictive model generated answers a determined question. We apply this model to a test case, which is a running process instance, and obtained a determined prediction, for example, if a failure risk is likely to occur or the predicted remaining time for a process instance in execution. Different predictive models are used in the literature. The most used are decision trees and association or decision rules due to their high level of interpretability.

**Knowledge deployment:** the use of the obtained information can be relevant for the decision-making during the execution of the process. The model generated and the predictions of performance insights can be else employed in the optimization and enhancement stages of the life-cycle of a process.

### 3 Conclusions

This paper describes a step-based architecture for the monitoring prediction of business process performance indicators. The work details all the stages of the methodology which are based on the KDD process. The application of this model can be considered for any generic process indicator.

### References

1. M. de Leoni, W.M.P. van der Aalst, and M. Dees. A general process mining framework for correlating, predicting and clustering dynamic behavior based on event logs. *Information Systems*, 56:235 – 257, 2016.
2. X. Liu, Y. Yang, D. Cao, and D. Yuan. Selecting checkpoints along the time line: A novel temporal checkpoint selection strategy for monitoring a batch of parallel business processes. In *IEEE ICSE Proceedings*, 2013.
3. C. Di Francescomarino, M. Dumas, F.M. Maggi, and I. Teinemaa. Clustering-based predictive process monitoring. *CoRR*, abs/1506.01428, 2015.
4. A. Bevacqua, M. Carnuccio, F. Folino, M. Guarascio, and L. Pontieri. A data-adaptive trace abstraction approach to the prediction of business process performances. In *Proceedings of the 15th International Conference on Enterprise Information Systems, Volume 1, ICEIS 2013*, pages 56–65, 2013.



# Verification and Validation of UML Artifact-centric Business Process Models<sup>\*</sup>

Montserrat Estañol<sup>1</sup>, Maria-Ribera Sancho<sup>1,2</sup>, and Ernest Teniente<sup>1</sup>

<sup>1</sup> Universitat Politècnica de Catalunya, Barcelona, Spain  
{estanyol|ribera|teniente}@essi.upc.edu

<sup>2</sup> Barcelona Supercomputing Center, Barcelona, Spain

## Resumen de la contribución

El framework BALSa proporciona una perspectiva para el modelado de procesos de negocio basada en cuatro dimensiones: artefactos, ciclos de vida, servicios y asociaciones [3]. Los artefactos sirven para definir los datos necesarios para llevar a cabo el proceso, los ciclos de vida representan la evolución de dichos artefactos, los servicios se corresponden a tareas, o unidades de trabajo, y las asociaciones limitan la forma en que los servicios modifican los artefactos.

Dicho framework se enmarca en el *enfoque centrado en artefactos (artifact-centric)*, diferenciable del *enfoque centrado en procesos (process-centric)*, cuyas representaciones se centran en mostrar el orden de ejecución de las tareas, sin dar importancia a los datos necesarios para llevarlas a cabo.

Una de las ventajas del enfoque *artifact-centric* respecto al enfoque *process-centric* es la posibilidad de determinar de forma automática la corrección semántica de dichos modelos, gracias a la definición de los datos y la especificación de las tareas necesarias para llevar a cabo el proceso. Nótese que, si no tenemos datos, no podemos definir exactamente qué hace cada una de las tareas, con lo cual el análisis no tiene en consideración el significado de éstas, y acaba siendo puramente sintáctico (errores en el uso del lenguaje) y/o estructural (por ej.: deadlocks). Si, además, realizamos este análisis antes del desarrollo y la implantación del proceso evitaremos la propagación de errores a su funcionamiento.

La mayoría de propuestas existentes que tratan con la corrección semántica de procesos *artifact-centric* están fundamentadas en el uso de lenguajes formales basados en la lógica. El principal problema de estas propuestas es que estos lenguajes son difíciles o incluso imposibles de comprender para las personas involucradas en el modelado y desarrollo del proceso de negocio.

Por este motivo, en este trabajo presentamos una propuesta para determinar potencialmente de forma automática la corrección semántica de un modelo *artifact-centric* definiendo el framework BAUML [2,1]. Dicho framework

<sup>\*</sup> Este artículo resume el trabajo publicado en la 27th Int. Conf. on Advanced Information Systems Engineering (CAiSE 2015). Esta conferencia está clasificada como CORE A y su porcentaje de aceptación en 2015 fue del 13,1%. Investigación financiada parcialmente por los proyectos TIN2011-24747-C2-2-R y 2014 SGR 1543.

se basa en el uso de una serie de modelos UML y OCL para representar las cuatro dimensiones definidas en el framework BALSa. Más concretamente, usamos el diagrama de clases para representar los artefactos, el diagrama de estados para los ciclos de vida, el diagrama de actividades para las asociaciones y los contratos de operaciones en OCL para mostrar los detalles de las tareas o servicios. Tanto UML y OCL son lenguajes estándar conocidos y usados en el ámbito empresarial, y más comprensibles que el uso de la lógica.

Para determinar la corrección de los modelos usamos una herramienta, SVTe, desarrollada por nuestro grupo de investigación. Dicha herramienta es capaz de determinar si un esquema en lógica de primer orden satisface una propiedad concreta, que especifica una característica deseable del proceso de negocio.

En aras de poder utilizar SVTe, en este trabajo se proponen distintos algoritmos para traducir los modelos BAUML a lógica de primer orden. Para lograrlo, se utiliza como punto de partida el trabajo realizado en [4], que consideraba únicamente el diagrama de clases UML y los contratos de las operaciones en OCL a la hora de llevar a cabo tests de corrección semántica. La traducción propuesta en [4] se modifica y se amplía aquí para añadir las restricciones impuestas por el diagrama de estados y los diagramas de actividades.

Además, distinguimos entre tests de verificación y tests de validación. Los tests de verificación proporcionan respuestas a la pregunta *¿es correcto el modelo?*, mientras que los tests de validación responden a *¿tenemos el modelo correcto?*. Los primeros buscan errores y contradicciones internas en el modelo, mientras que los segundos comprueban que el modelo represente correctamente los requisitos impuestos por el dominio. Por este motivo, los tests de verificación no requieren interpretación del resultado por parte del ingeniero de negocio y pueden realizarse automáticamente a partir del modelo BAUML inicial, mientras que los segundos mayoritariamente deberán ser generados por el usuario y los resultados requerirán de su interpretación.

La propuesta aquí presentada es la primera que considera conjuntamente todos los diagramas UML y los contratos OCL de BAUML en el razonamiento. La factibilidad del enfoque se demuestra mediante una herramienta existente, SVTe, que proporciona resultados a partir de la traducción de los modelos BAUML.

## Referencias

1. Calvanese, D., Montali, M., Estañol, M., Teniente, E.: Verifiable UML artifact-centric business process models. In: Li, J., Wang, X.S., Garofalakis, M.N., Soboroff, I., Suel, T., Wang, M. (eds.) CIKM 2014. pp. 1289–1298. ACM (2014)
2. Estañol, M., Queralt, A., Sancho, M.R., Teniente, E.: Specifying artifact-centric business process models in UML. In: Shishkov, B. (ed.) BMSD 2014, Revised Selected Papers. LNBIP, vol. 220, pp. 62–81. Springer (2015)
3. Hull, R.: Artifact-centric business process models: Brief survey of research results and challenges. In: Meersman, R., Tari, Z. (eds.) OTM 2008. LNCS, vol. 5332, pp. 1152–1163. Springer (2008)
4. Queralt, A., Teniente, E.: Reasoning on UML conceptual schemas with operations. In: van Eck, P., Gordijn, J., Wieringa, R. (eds.) CAiSE 2009. pp. 47–62. LNCS, Springer (2009)

# Automating Correctness Verification of Artifact-Centric Business Process Models\*

Diana Borrego, Rafael M. Gasca, and María Teresa Gómez-López

Department of Computer Languages and Systems, University of Seville, Spain  
{dianabn, gasca, maytegozmez}@us.es  
<http://www.idea.us.es/>

## 1 Resumen

Traditionally, business processes are modelled as activity-centric business process models in which activities are focused on and data just serve as inputs and outputs of some services. They follow the imperative principles, implying that the workflow of the activities can be defined at design time. But for some types of problems, it is easier to represent how the data are modified during the process execution instead of the activities that execute the data evolution.

For this reason, the artifact-centric methodology (data-centric approach) has emerged as a new paradigm to support business process management, where business artifacts appeared for the necessity of enrich the business process model with information about data providing a way for understanding the interplay between data and process. Artifacts are business-relevant objects that are created, evolved, and (typically) archived as they pass through a business, combining both data aspects and process aspects into a holistic unit.

Artifact-centric modelling establishes data objects (called artifacts) and their lifecycles as focus of the business process modelling. This type of modelling is inherently declarative: the control flow of the business process is not explicitly modelled, but follows from the lifecycles of the artifacts.

The lifecycle represents how the state of an artifact may evolve over the time. The different activities change the state of the artifact and the values of the data associated to each artifact; these may be manual or automatic. The evolution of the artifacts implies a change of the state and the values of the data, until a goal state of an artifact is reached. One of the reasons why the artifact-centric paradigm facilitates the process description is the capacity to model the relations between objects with different cardinalities, not only 1-to-1 relations. This modeling capabilities are not entirely supported in activity-centric scenarios. For instance, BPMN 2.0 (currently wide accepted activity-centric notation) allows to easily represent multi-instance activities and pools (processes), but with some

---

\* Original paper: Automating Correctness Verification of Artifact-Centric Business Process Models. Published in Journal of Information and Software Technology (IF: 1.046, Q2 in two categories). Vol. 62, Issue C, Pages 187-197, June 2015. DOI: 10.1016/j.infsof.2015.02.010

limitations: (i) relations between different processes can only be expressed as hierarchies, where a process can invoke multiple instances of its subprocess; (ii) the return value of an executed sub-process instance is only accessible when its execution finishes, not allowing the interaction of another process during the execution; and (iii) the definition of Data Objects, Data Inputs, Data Outputs, Sets, and Data Associations in BPMN 2.0 allows to specify collections of elements, but it does not permit data instance differentiation or to include the relation between the data objects between them.

When more than one artifact is involved in the process, it is possible that a combination of services and data values violate the policies of the business. In order to avoid this situation at runtime, it is possible to detect some of these possible errors even at design time. Specifically, the errors derived from an incorrect design of the model. In spite of the unknown runtime data in the design time phase, our proposal is able to perform a data verification of the models by means of the use of mandatory domains of values, which can be obtained from previous executions and/or knowledge from experts. Making use of this information, it is possible to determine the existence of certain errors in the structural and data perspectives of the model before it is deployed.

The paper develops a fully automated approach for verifying the correctness of artifact-centric models at design time, including the state relation between the artifacts of the model (handling 1-to-N and N-to-M associations between artifacts), and the data values that define the relations between them. To develop the automatic verification, services are formally modeled using pre and postconditions over the data associated to the artifacts' states. To analyse the correctness of the model, it is necessary to study when the services can be executed. A service can be executed if the evolution of the lifecycle of the artifact arrives at the service and its precondition is satisfied. Upon completion, the service delivers data that satisfies its postconditions. The non-satisfiability of a pre or postcondition can cause that the lifecycle gets stuck at a service and fails.

Two correctness notions are distinguished (reachability and weak termination), and novel verification algorithms are developed to check them. In more detail, our proposal consists of various phases: (a) preprocessing is applied to detect basic data anomalies; (b) the artifact union graph is translated into a CSP formulation in order to automatize the verification, avoiding its manual performance which is time-consuming and error-prone; (c) the CSP formulation models the lifecycle of the artifacts with pre and postconditions of the services and the numerical data managed, analysing the possible interaction among the different artifacts; and (d) in case of an error, feedback is provided by determining the states which are not reachable and/or weak-terminable. The approach is complete, so it always generates accurate feedback in case of an error.

To the best of our knowledge, this paper presented the first verification approach for artifact-centric business process models that integrates pre and postconditions, which define the behavior of the services, and numerical data verification when the model is formed of more than one artifact. The approach can detect errors not detectable with other approaches.

# Process Mining: Past, Present and (Likely) Future

Josep Carmona

Universitat Politècnica de Catalunya, Barcelona (Spain)  
jcarmona@cs.upc.edu

**Abstract.** The process mining field came here to stay. This is demonstrated by the growing interest in the last decade, both in academia and industry. Being at the intersection of many disciplines, process mining techniques transform structured information into valuable models, which provide a fresh and formal insight into the real execution of processes within an organization. Still, there is way more to do than what has been accomplished: *process discovery techniques* may suffer from noisy, incomplete event logs and may fail to choose the right representational bias; *conformance checking* suffers from the inherent complexity of working at the state-space level, and in case of large inputs this fact prevents from *enhancing* process models with additional perspectives. In this paper I will provide an historical overview of the field, describe its current challenges, and vaticinate its long-term future.

## 1 Introduction

Our daily life is full of data stored by information systems in form of event logs. These logs contain the footprints left by process executions, e.g., “the hiring process”, “the surgery process”, “the repair process”, etc. Quite often happens that the corresponding models specifying these processes are not well-specified or simply out of date, thus hampering the formal analysis one can do in the underlying system through the process model (e.g., analyze the performance). *Process mining* is an emerging field that focuses on analyzing these event logs with the purpose of extracting, analyzing and enhancing evidence-based process models [23].

In this paper we provide an informal and personal overview of the current situation of the field. The following sections will report the recent history of the field, and the current situation and future of the main two dimensions: process discovery and conformance checking.

## 2 Process Discovery

The discovery of process models from system logs was firstly suggested in [9], in the scope of software engineering. In that work, algorithms for the derivation of a transition system that represents the traces in the event log obtained from

software processes were presented. Being the most popular option, we will focus in this section in the discovery of Petri nets. The first method to obtain a Petri net from an event log was presented in [26]. The method (called  $\alpha$ -algorithm) was based on detecting the causal relations in the traces and constructing the Petri net based on these relations. The algorithm derives nets within a particular class of Petri nets called *Structured Workflow Nets*, and the technique proved to be a very simple manner to obtain a Petri net from an event log. Several algorithms have been presented in the literature to overcome the inability of the  $\alpha$ -algorithm in capturing certain behaviors (e.g. [31,5,27,16]). Also, evolutionary approaches have been presented [25,4,30], being able to extend the class of Petri nets obtained and which also considered event logs containing noise. Finally, clustering methods have also been investigated [14,11].

The following is the list of topics not yet resolved which represent a challenge for the widespread application of process discovery techniques:

- Log Incompleteness: the log only contains a portion of the traces of the underlying process.
- Representational Bias: the pre-selection of the formalism to represent the observed behavior may hinder the quality of the obtained result.
- Noise: traces in the event log may not always represent the underlying process.
- Mining Data: only mature algorithms exist for the *control-flow*, whilst the rest of available data is only used a-posteriori.
- Spagnetti Models: although the right representation bias may be chosen, still complex behavior requires often cluttered models that do not offer any interesting insight on the underlying process (apart from looking like a mess).
- Algorithms Interface: current discovery techniques need to be configured by assigning several parameters. A wrong selection on the parameters may seriously degrade the performance of the algorithm.
- Concept Drift: event logs are not static objects, but dynamic. This means that several process may be stored in an event log (think of a shop with summer/winter sales). Having several instances that correspond to different processes is a problem for current discovery techniques.

### 3 Conformance Checking

Conformance checking techniques aim to assess the adequacy of a model for describing observed behavior, as manifested in an event log. Four dimensions were envisioned for this task [21]:

- Fitness** quantifies how much of the observed behavior is captured by the model,
- Precision** quantifies how much behavior exists in the model that was not observed in the event log,
- Generalization** quantifies how well the model explains behavior of the system that was not observed in the log, and
- Simplicity** quantifies the complexity of the model.

For measuring these dimensions, it is crucial to relate the behavior of the model with the traces of the log. The seminal work by [21] represents the first attempt to relate observed and modeled behavior. Given an observed trace, it is based on heuristically replaying it on the process model. Although in practice this approach may be very useful for large models, for indeterministic models it cannot guarantee the existence of a solution even though it exists. Evolutions of this replay technique can also be found in recent work [28,29], which also inherit the aforementioned fundamental problem of replaying techniques.

The work in [1] proposed the notion of *alignment* for the first time, and developed a technique to compute optimal alignments for a particular class of process models. Intuitively, given a trace representing a real process execution, an optimal alignment provides the best trace the process model can provide to mimic the observed behavior. Alignments are extremely important in the context of process mining, since they open the door to evaluate the metrics that assess the quality of a process model to represent observed behavior: *fitness* and *generalization* [3,1] and precision [2]. Additionally, alignments are a necessary step to enhance the information provided in a process model [23].

Decompositional techniques have been recently presented [24,19] that instead of computing optimal alignments, they focus on the *decisional problem* of deciding whereas a given trace fits or not a process model. The underlying idea is to split the model into a particular set of transition-bordered fragments which satisfy certain conditions, and local alignments can be computed for each one of the fragments, thus providing an upper bound on the cost of an alignment.

Finally, the work in [18,17] considers the problem of dealing with partially ordered event data, a common situation in certain context like health care. The notion of *partially ordered alignment* is introduced, and a variation of the techniques presented in [1] is used for its computation.

Hence, the most important challenge in conformance checking is the computation of an alignment. Unfortunately, the current algorithmic support to compute alignments is defined as search for a minimal path on the product of the state space of the process model and the observed behavior, an object that is worst-case exponential with respect to the size of the model. This hampers the application of these techniques for medium/large instances.

## 4 The (Likely) Future

*Discovery.* Apart from the topics listed in Section 2, some important research challenges will be faced in the following years with respect to the discovery of process models. First, the full discovery of the widespread BPMN notation will receive attention [8]. Second, discovery techniques that incorporate the efficient handling of massive data and/or connect to data warehouses will be investigated [15,12]. Third, the incorporation of new dimensions or perspectives in the discovery problem (e.g., artifacts [20]) will be considered.

*Conformance Checking.* As commented in Section 3, efficient approaches for conformance checking (and in particular, alignment computation) are yet to

come [22]. As for the case of discovery, techniques that compute alignments on top of BPMN will also be seen in the following years. Also, the proposal of mature techniques for precision and generalization will expand the usage of conformance checking in practice [7]. Finally, extending conformance checking to not only focus in control-flow but other dimensions will be considered [13,10].

*Tool Support.* The current tool support for process mining is not sufficient for satisfying the expectations the discipline has created. On the one hand, ProM, the de-facto tool in the area, is seen like a very complex tool even for process mining practitioners. Recently, ProM Lite has appeared, which simplifies significantly the interaction with the tool. A more scripting arena, much in the style of Matlab, would complement the capabilities of a visual tool like ProM. On this direction, tools like PMLAB [6] will be developed in the near future, offering a flexible environment for scripting process mining techniques.

*Industry.* In the last decade there has been several industrial initiatives for process mining products: Fujitsu, Fluxicon, QPR Software, Celonis, Gradient, among others. The tendency is to focusing on the discovery of simple process models (e.g., fuzzy models in Disco from Fluxicon). In the next decade we will not only see more of these initiatives, but also industrial initiatives for the problem of conformance checking, a clear missing in the industrial sector nowadays.

**Acknowledgments.** This work has been partially supported by funds from the Spanish Ministry for Economy and Competitiveness (MINECO), the European Union (FEDER funds) under grant COMMAS (ref. TIN2013-46181-C2-1-R).

## References

1. Arya Adriansyah. *Aligning observed and modeled behavior*. PhD thesis, Technische Universiteit Eindhoven, 2014.
2. Arya Adriansyah, Jorge Munoz-Gama, Josep Carmona, Boudewijn F. van Dongen, and Wil M. P. van der Aalst. Measuring precision of modeled behavior. *Inf. Syst. E-Business Management*, 13(1):37–67, 2015.
3. Arya Adriansyah, Boudewijn F. van Dongen, and Wil M. P. van der Aalst. Conformance checking using cost-based fitness analysis. In *EDOC*, pages 55–64. IEEE Computer Society, 2011.
4. Joos C. A. M. Buijs, Boudewijn F. van Dongen, and Wil M. P. van der Aalst. A genetic algorithm for discovering process trees. In *Proceedings of the IEEE Congress on Evolutionary Computation, CEC 2012, Brisbane, Australia, June 10-15, 2012*, pages 1–8, 2012.
5. Josep Carmona, Jordi Cortadella, and Michael Kishinevsky. A region-based algorithm for discovering Petri nets from event logs. In M. Dumas, M. Reichert, and M. C. Shan, editors, *BPM*, volume 5240 of *Lecture Notes in Computer Science*, pages 358–373. Springer, 2008.
6. Josep Carmona and Marc Solé. PMLAB: an scripting environment for process mining. In *Proceedings of the BPM Demo Sessions 2014 Co-located with the 12th International Conference on Business Process Management (BPM 2014)*, Eindhoven, The Netherlands, September 10, 2014., page 16, 2014.



7. Thomas Chatain and Josep Carmona. Anti-alignments in conformance checking - the dark side of process models. In *Application and Theory of Petri Nets and Concurrency - 37th International Conference, PETRI NETS 2016, Toruń, Poland, June 19-24, 2016. Proceedings*, pages 240–258, 2016.
8. Raffaele Conforti, Marlon Dumas, Luciano García-Bañuelos, and Marcello La Rosa. BPMN miner: Automated discovery of BPMN process models with hierarchical structure. *Inf. Syst.*, 56:284–303, 2016.
9. Jonathan E. Cook and Alexander L. Wolf. Discovering models of software processes from event-based data. *ACM Trans. Softw. Eng. Methodol.*, 7(3):215–249, 1998.
10. Massimiliano de Leoni and Wil M. P. van der Aalst. Aligning event logs and process models for multi-perspective conformance checking: An approach based on integer linear programming. In *Business Process Management - 11th International Conference, BPM 2013, Beijing, China, August 26-30, 2013. Proceedings*, pages 113–129, 2013.
11. A.K. Alves de Medeiros, A. Guzzo, G. Greco, W.M.P. van der Aalst, A.J.M.M. Weijters, B. van Dongen, and D. Sacca. Process mining based on clustering: A quest for precision. In B. Benatallah A. ter Hofstede and H.Y. Paik, editors, *BPM 2007 International Workshops (BPI, BPD, CBP, ProHealth, RefMod, Semantics4ws)*, volume 4928 of *Lecture Notes in Computer Science*, pages 17–29. Springer, 2008.
12. Eduardo González López de Murillas, Hajo A. Reijers, and Wil M. P. van der Aalst. Connecting databases with process mining: A meta model and toolset. In *Enterprise, Business-Process and Information Systems Modeling - 17th International Conference, BPMDS 2016, 21st International Conference, EMMSAD 2016, Held at CAiSE 2016, Ljubljana, Slovenia, June 13-14, 2016. Proceedings*, pages 231–249, 2016.
13. Dirk Fahland, Massimiliano de Leoni, Boudewijn F. van Dongen, and Wil M. P. van der Aalst. Behavioral conformance of artifact-centric process models. In *Business Information Systems - 14th International Conference, BIS 2011, Poznań, Poland, June 15-17, 2011. Proceedings*, pages 37–49, 2011.
14. Gianluigi Greco, Antonella Guzzo, Luigi Pontieri, and Domenico Sacca. Discovering expressive process models by clustering log traces. *IEEE Trans. Knowl. Data Eng.*, 18(8):1010–1027, 2006.
15. Sander J. J. Leemans, Dirk Fahland, and Wil M. P. van der Aalst. Scalable process discovery with guarantees. In *Enterprise, Business-Process and Information Systems Modeling - 16th International Conference, BPMDS 2015, 20th International Conference, EMMSAD 2015, Held at CAiSE 2015, Stockholm, Sweden, June 8-9, 2015. Proceedings*, pages 85–101, 2015.
16. Sander J. J. Leemans, Dirk Fahland, and Wil M. P. van der Aalst. Discovering block-structured process models from incomplete event logs. In Gianfranco Ciardo and Ekkart Kindler, editors, *Application and Theory of Petri Nets and Concurrency*, volume 8489 of *Lecture Notes in Computer Science*, pages 91–110. Springer International Publishing, 2014.
17. Xixi Lu, Dirk Fahland, and Wil M. P. van der Aalst. Conformance checking based on partially ordered event data. In *Business Process Management Workshops - BPM 2014 International Workshops, Eindhoven, The Netherlands, September 7-8, 2014, Revised Papers*, pages 75–88, 2014.
18. Xixi Lu, Ronny Mans, Dirk Fahland, and Wil M. P. van der Aalst. Conformance checking in healthcare based on partially ordered event data. In *Proceedings of the 2014 IEEE Emerging Technology and Factory Automation, ETFA 2014, Barcelona, Spain, September 16-19, 2014*, pages 1–8, 2014.

19. Jorge Munoz-Gama, Josep Carmona, and Wil M. P. van der Aalst. Single-entry single-exit decomposed conformance checking. *Inf. Syst.*, 46:102–122, 2014.
20. Viara Popova, Dirk Fahland, and Marlon Dumas. Artifact lifecycle discovery. *Int. J. Cooperative Inf. Syst.*, 24(1), 2015.
21. Anne Rozinat and Wil M. P. van der Aalst. Conformance checking of processes based on monitoring real behavior. *Information Systems*, 33(1):64–95, 2008.
22. Farbod Taymouri and Josep Carmona. A recursive paradigm for aligning observed behavior of large structured process models. In *To appear at Business Process Management Conference (BPM), Rio de Janeiro, Brazil, September 2016.*, Lecture Notes in Computer Science. Springer.
23. Wil M. P. van der Aalst. *Process Mining - Discovery, Conformance and Enhancement of Business Processes*. Springer, 2011.
24. Wil M. P. van der Aalst. Decomposing Petri nets for process mining: A generic approach. *Distributed and Parallel Databases*, 31(4):471–507, 2013.
25. Wil M. P. van der Aalst, Ana Karla A. de Medeiros, and A. J. M. M. Weijters. Genetic process mining. In *Applications and Theory of Petri Nets 2005, 26th International Conference, ICATPN 2005, Miami, USA, June 20-25, 2005, Proceedings*, volume 3536 of *Lecture Notes in Computer Science*, pages 48–69. Springer, 2005.
26. Wil M. P. van der Aalst, Ton Weijters, and Laura Maruster. Workflow mining: Discovering process models from event logs. *IEEE Trans. Knowl. Data Eng.*, 16(9):1128–1142, 2004.
27. Jan Martijn E. M. van der Werf, Boudewijn F. van Dongen, Cor A. J. Hurkens, and Alexander Serebrenik. Process discovery using integer linear programming. In Kees M. van Hee and Rüdiger Valk, editors, *Petri Nets*, volume 5062 of *Lecture Notes in Computer Science*, pages 368–387. Springer, 2008.
28. Seppe K. L. M. vanden Broucke, Jorge Munoz-Gama, Josep Carmona, Bart Baesens, and Jan Vanthienen. Event-based real-time decomposed conformance analysis. In *On the Move to Meaningful Internet Systems: OTM 2014 Conferences - Confederated International Conferences: CoopIS, and ODBASE 2014, Amantea, Italy, October 27-31, 2014, Proceedings*, pages 345–363, 2014.
29. Seppe K. L. M. vanden Broucke, Jochen De Weerd, Jan Vanthienen, and Bart Baesens. Determining process model precision and generalization with weighted artificial negative events. *IEEE Trans. Knowl. Data Eng.*, 26(8):1877–1889, 2014.
30. Borja Vázquez-Barreiros, Manuel Mucientes, and Manuel Lama. A genetic algorithm for process discovery guided by completeness, precision and simplicity. In *Business Process Management - 12th International Conference, BPM 2014, Haifa, Israel, September 7-11, 2014. Proceedings*, pages 118–133, 2014.
31. Lijie Wen, Wil M. P. van der Aalst, Jianmin Wang, and Jianguang Sun. Mining process models with non-free-choice constructs. *Data Min. Knowl. Discov.*, 15(2):145–180, 2007.

Ecosistemas de servicios, personas y cosas



## Reconfiguration of Service Failures in DAMASCo using Dynamic Software Product Lines\*

Javier Cubo, Nadia Gamez, Ernesto Pimentel, Lidia Fuentes

Dpto de Lenguajes y Ciencias de la Computación, Universidad de Málaga  
{cubo, nadia, ernesto, lff}@lcc.uma.es

**Abstract.** Building service-based applications requires providing the ability to handle, maintain or upgrade the services that compose these applications. As same services may be used by a wide variability of applications, the management of the heterogeneity at runtime is required. This is crucial to reconfigure applications in case of service failures. The DAMASCo framework reduces the complexity of modeling services focusing on the discovery, composition and adaptation of context-aware services. But currently, it does not support the dynamic reconfiguration of service-based applications. In this work, we follow a Dynamic Software Product Line approach to extend DAMASCo for providing reconfiguration to support specific situations of fails at runtime. We propose a novel approach of grouping services in families facilitating the selection and usage of similar services in case of fails. We apply our approach to an intelligent transportation system case study where DAMASCo composes and reconfigure the necessary services to provide a dynamic route for a driver's request.

**Keywords:** Service Reconfiguration; Heterogeneity; DSPL.

## Using Dynamic Software Product Lines for Reconfiguration

Service-based applications need infrastructures with reconfiguration capabilities, to meet the changing requirements of businesses, environments and users. These infrastructures must handle, maintain or upgrade the services that compose these applications. In general, Service-Oriented Architectures (SOA) promotes the reusability and interoperability of entities. In this sense, DAMASCO is a framework based on SOA, which reduces the complexity of modeling services focusing on the reuse of services accessed via their public interfaces. DAMASCO discovers the most appropriate services to satisfy a user's request considering context information and semantic matchmaking, and, if necessary, adapts them during the composition with the purpose of performing the required functionalities. These services could serve to several applications running at the same time. Once services are composed and are being executed, failures may occur at runtime. This requires proper reactions to dynamically reconfigure the applications using the services involved in the failures. These applications are very heterogeneous and then the requirements for reconfiguration can be variable.

Software Product Line (SPL) is a very well-known approach that promotes the reuse of the core asset of a family of products, by managing the heterogeneity. At first, following a SPL approach, we extended DAMASCO to deal with the service variability during the service composition. We used Feature Models (FMs) to model the service variability at design and composition time, introducing the concept of service family. A service family includes services with similar functionalities, but slightly different at design level (e.g., services whose interfaces have distinct internal activities to implement the same functionalities).

When services fail, the management of the heterogeneity only at composition time is not enough. Then, it also needs to be managed at runtime to allow the reconfiguration of heterogeneous applications. In order to address this need, we propose to use Dynamic Software Product Lines (DSPLs), which produce software products capable of being adapted to environmental changes at runtime. In this work, the changes we consider are contingent upon whether a service is operating or not (because a failure), and the products to be adapted are the service-based applications using the services involved in the failures. Therefore, in this paper we present the extension of the DAMASCO framework, which currently does not support the service

---

\* Work published in *12th IEEE International Conference on Services Computing (SCC 2015)*. It is supported by projects TIN2012-35669 and TIN2012-34840, funded by Spanish Ministry MINECO, FEDER; P11-TIC-7659, P09-TIC-5231 and P12-TIC1814 funded by Andalusian Gov; FP7-610531 and FP7-317731 funded by EU; and Univ. Málaga, Campus Exc. Int. Andalucía Tech.

failures at runtime, with a dynamic reconfiguration process driven by Feature Models based on DSPLs. Therefore, we follow a DSPL-based approach to reconfigure dynamically failed services, as shown in Figure 1.

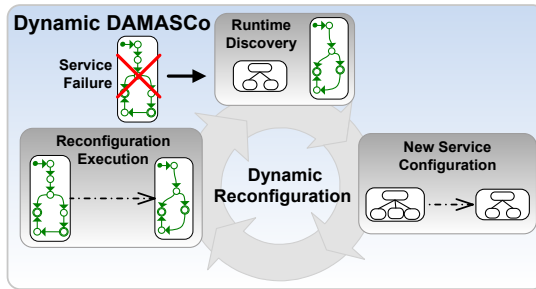


Fig. 1. Reconfiguration in Dynamic DAMASCo

The advantages of our proposal, Dynamic DAMASCo, are the following:

- Services are grouped in families represented by a feature model. When a service fails, we will search for another service from the same family, avoiding a new discovery.
- Application reconfiguration requirements are variable, due to the heterogeneity of the applications. Then, although several applications are using the same service, when this fails, reconfiguration actions may be different for each application to satisfy its specific requirements. We deal with this aspect by using the feature models that allows finding different services for distinct requirements.
- Compose and upgrade on-the-fly services selecting concrete internal activities of other services of the family, modeled in the same feature model. This enables that when a service fails, it does not need to be replaced for a new one, instead, we can replace just some activities.

Then, the main contributions compared to previous efforts are focused not only on identifying when and how reconfiguring services, but also injecting code to replace failed services; and exploiting the service family and feature models to handle the heterogeneity of the application.

## Service Reconfiguration with Dynamic DAMASCo

Following our approach, depicted in Figure 1, after a service failure occurs, the first step of this process is the **Runtime Discovery**, which finds, for every application affected for the failure, a new service to replace the failed service. The new service is a member of the service family of the failed service, which will be selected taking into account the different application requirements and contexts. Then, the regular discovery process does not need to be performed again. In order to perform this selection, Dynamic DAMASCo uses Hydra to generate, from the feature model representing the service family, the service configuration that satisfies the needs of every application. Then, the use of feature models also helps to manage the heterogeneity of the application requirements. The new service configuration can be different for each request where the failed service is involved. Our approach also selects the service with the maximum number of equivalent internal activities with respect to the failed service. The rationale behind of this is to be able to replace on-the-fly just some activities instead of replacing the whole service.

Then, once we have the two feature model configurations, the one that represents the service previously being executed, and the one representing the new service, we need to determine the way to achieve the **New Service Configuration**. In this step, detailed in Section IV.B, we obtain, at model level, the exact changes to be performed to reconfigure each application that was using the failed service. These specific modifications are calculated from the difference between the two configurations.

Finally, the modifications specified at model level have to be carried out at implementation level. This **Reconfiguration Execution** is performed adding and/or removing service internal activities or full services. This injection of code is done using runtime monitoring and binding processes, detailed in Section IV.C.



# Automatización de la etapa de análisis para la aplicación de la técnica de prueba metamórfica a composiciones de servicios WS-BPEL

M Carmen de Castro-Cabrera, Kevin J Valle-Gómez, Inmaculada Medina-Bulo

Departamento de Ingeniería Informática, Universidad de Cádiz  
Avda. Universidad de Cádiz, 10, 11519, Puerto Real, España  
kevin.vallegomez@alum.uca.es,  
{maricarmen.decastro,inmaculada.medina}@uca.es

**Resumen** La importancia de los servicios web en el mundo actual está en auge, y el impacto social que están teniendo gracias a las nuevas tecnologías desarrolladas, hace necesario el diseño de mejores técnicas de prueba para el software específico de composición de estos servicios web. Una de las técnicas propuestas para las composiciones WS-BPEL es la técnica de prueba metamórfica. En trabajos anteriores se ha presentado una arquitectura para su aplicación y se han aportado algunas ideas para la automatización de la etapa inicial de la misma, que se corresponde con el análisis y obtención de propiedades, pero no se llegaron a desarrollar ni implementar. En este trabajo se presenta el diseño de una solución para automatizar ciertos aspectos de la etapa de análisis y obtención de propiedades, la cual ha sido probada en diferentes casos de prueba obteniéndose buenos resultados.

**Keywords:** pruebas metamórficas, relaciones metamórficas, WS-BPEL, composiciones de servicios web, solución, análisis

## 1. Introducción

El lenguaje *WS-BPEL 2.0* [5] permite, a partir de servicios web (WS) ya existentes, crear procesos de negocio complejos. A pesar de la rápida y continua mejora de este tipo de servicios, las técnicas de prueba empleadas para probarlos no han avanzado al mismo ritmo. Por ello, es de máximo interés investigar en esta línea e implementar nuevos métodos de pruebas en este campo, así como, mejorar los existentes.

La *técnica de prueba metamórfica* (PM) [3], que ha sido aplicada con éxito en lenguajes de programación tradicionales [7], permite verificar el software obteniendo nuevos casos de prueba a partir de otros iniciales mediante la aplicación de las denominadas relaciones metamórficas (RM). Las RM son propiedades esperadas o existentes en el software, definidas sobre un conjunto de entradas y salidas.

El *análisis y obtención de propiedades* de las composiciones WS-BPEL 2.0 es uno de los aspectos más críticos en la arquitectura propuesta en [2]. Debido a la

complejidad de estas composiciones y las reglas de negocio existentes, más allá de los procesos puramente matemáticos, esta tarea se realiza normalmente de manera manual, creando una clara necesidad de automatización con el objetivo de agilizar el proceso y los costes.

En este trabajo se describe un proceso automatizado para la fase de análisis para su posterior integración en la arquitectura propuesta en [2]. La información obtenida en el proceso de análisis será utilizada para la implementación de RMs. Este trabajo se estructura de la siguiente forma: En la sección 2 se expondrán los fundamentos necesarios para abordar el resto del trabajo. En la sección 3 se describe proceso seguido para el análisis de las composiciones WS-BPEL, mientras que en la sección 4 se muestra un caso de estudio. Finalmente, se detallan las conclusiones y el trabajo futuro en la sección 5.

## 2. Fundamentos

A continuación se describen los fundamentos, en este orden: el lenguaje WS-BPEL, la técnica prueba metamórfica y las herramientas utilizadas.

*WS-BPEL* [5] está basado en XML y permite especificar el comportamiento de un proceso de negocio basado en sus interacciones con servicios web (SW). Se utiliza para generar nuevos procesos de negocios a partir de servicios ya existentes. Los bloques principales que forman las composiciones WS-BPEL son las *actividades*.

El concepto de PM está muy relacionado con las RM. Una definición para RM según [1] sería *propiedades existentes sobre el conjunto de entradas y sus correspondientes resultados para múltiples evaluaciones de una función*; es decir, siempre que la entrada sea correcta, con las RM esperamos una salida que cumpla determinadas propiedades necesarias. Una RM debe proveer una forma de generar nuevos casos de prueba a partir de otros ya existentes.

El desarrollo de la arquitectura propuesta en [2] requiere la consecución de varias etapas, cuya interacción permite la aplicación de PM. Para la etapa de análisis y obtención de propiedades, es necesario un analizador de composiciones capaz de obtener de forma automática información relevante que facilite la elección e implementación de las propiedades a aplicar a cada composición.

Para la implementación se hace uso de algunos módulos de la herramienta *test-generator-autoseed* [4], y de la utilidad *XMLStarlet*, que permite modificar y validar XML. Su uso será descrito en los siguientes apartados.

## 3. Análisis y extracción de información

Para realizar el proceso de análisis de composiciones de la arquitectura [2] con la mínima intervención humana posible, se propone la implementación de una solución compuesta por los siguientes pasos:

1. Desarrollo de una aplicación capaz de utilizar mediante herencia funciones predefinidas en la herramienta **test-generator-autoseed** [4] para obtener

- información detallada de los elementos presentes en la composición WS-BPEL. Así mismo, se dispondrá de otro módulo de análisis que mediante el uso de **XMLStarlet** y funciones XPath obtendrá información detallada de los elementos presentes en los casos de prueba.
2. Puesta en común de los resultados obtenidos en cada parte, con el objetivo de determinar qué elementos serán más propensos a conducir a la obtención de RM.
  3. Presentación de un informe en dos formatos, uno con formato técnico y otro escrito en lenguaje natural, destinados a la máquina y el analista, respectivamente.

La herramienta **test-generator-autoseed** provee de un conjunto de funciones *getters* y *setters* que serán reutilizados para la extracción de información de los elementos presentes en las composiciones WS-BPEL. Así mismo, mediante el uso de expresiones XPath junto a **XMLStarlet**, podemos explorar el contenido de sus casos de prueba.

#### 4. Aplicación a un caso de estudio

Para mostrar el resultado, se utilizará la composición *LoanApproval* [6]. Esta composición consiste en un proceso de aprobación de préstamos a partir de una cantidad monetaria, cuyo límite fijado en 10.000 unidades monetarias, determinará la acción de un aprobador de préstamos y/o un asesor financiero, como puede observarse en la Figura 1.

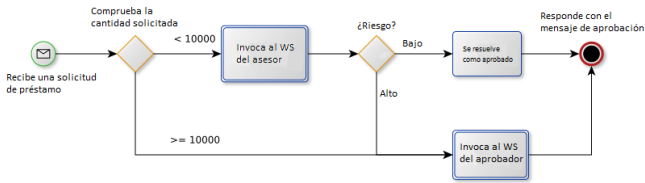


Figura 1. Lógica de la composición LoanApproval

La solución propuesta generará un informe disponible en dos formatos:

1. Una serie de ficheros CSV, cuyo formato genérico permitirá su gestión automática para la obtención de RM y otros usos que en su momento se crean convenientes. Un ejemplo del fichero correspondiente a las actividades IF de la composición *LoanApproval* puede verse en la Figura 2.
2. Un informe en lenguaje natural, que se utilizará de manera auxiliar para facilitar la tarea del análisis manual en el caso de que este fuese necesario.

type	name	condition	else
IF	lf1	( number(string(\$processInput.input/ns0:amount) ) &lt;= 10000 )	yes
IF	lfLowRisk	( \$risk.level = 'low' )	yes

**Figura 2.** Fragmento CSV

Se han obtenido manualmente y probado con buenos resultados algunas RM de la composición *LoanApproval*. Se detalla a continuación una de ellas:

$$\text{RM1: } req\_amount_1 \cdot 10 > 10,000 \wedge req\_amount_2 = req\_amount_1 \cdot 10 \wedge ap\_reply_2 = not(ap\_reply_1) \wedge as\_reply_2 = as\_reply_1 \implies accepted_2 = not(accepted_1) \wedge ap\_reply_2 = accepted_2$$

Los elementos presentes en ella, se obtienen de la información contenida en la composición WS-BPEL y sus casos de prueba. Concretamente, podemos observar que la constante 10.000 es la cantidad límite que aparece en la lógica del programa, junto al operador relacional  $>$ , que determina qué rama debe seguirse en la ejecución de la composición, implica que debe haber un elemento condicional IF. Estos elementos aparecen en el ejemplo mostrado en la Sección 4, lo que implica que la información obtenida se considera de utilidad para la obtención de RM.

## 5. Conclusiones y trabajo futuro

Los procesos de negocio que se basan en WS-BPEL aumentan cada año, creando la necesidad de mejores técnicas de prueba para este tipo de software, obligando a adaptar técnicas tradicionales para verificar estas composiciones.

En este trabajo se ha presentado una aplicación capaz de automatizar parte de la etapa de análisis de composiciones de la arquitectura propuesta en [2], para su integración en la misma.

Como trabajo futuro, está completar la implementación de esta aplicación: Para ello, se está trabajando, en la obtención otros tipos de formatos de ficheros enfocados a la automatización del tratamiento de los resultados, haciendo la solución más fácilmente legible por un programa. Por otro lado, se trabajará en la mejora de la salida en lenguaje natural.

## 6. Agradecimientos

Este trabajo ha sido financiado por el Programa Nacional para la Investigación, Desarrollo e Innovación del MICINN, proyecto DARDOS (TIN2015-65845-C3-3-R).

## Referencias

1. Andrews, J.H., Briand, L.C., Labiche, Y.: Is mutation an appropriate tool for testing experiments? In: Proceedings of the 27th International Conference on Software Engineering (ICSE 2005). pp. 402–411. ACM Press (2005)
2. Castro-Cabrera, M.d.C., Camacho-Magriñán, A., Medina-Bulo, I., Palomo-Duarte, M.: Una arquitectura basada en pruebas metamórficas para composiciones de servicios ws-bpel. In: Actas de las VII Jornadas de Ciencia e Ingeniería de Servicios. pp. 9–22. Servizo de publicacións da Universidade da Coruña, A Coruña, España (Sep 2011)
3. Chen, T.Y.: Metamorphic testing: A new approach for generating next test cases. HKUSTCS98-01 (1998)
4. Liñeiro Barea, V.: Herramienta para la generación automática de casos de prueba mediante siembra automática para WS-BPEL 2.0 (feb 2014), <http://rodin.uca.es:80/xmlui/handle/10498/16222>
5. OASIS: Web Services Business Process Execution Language 2.0. <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html> (2007), Organization for the Advancement of Structured Information Standards
6. UCASE Research Group: LoanApprovalDoc - WS-BPEL Composition Repository - Redmine ((accessed Jun 30, 2016)), <https://neptuno.uca.es/redmine/projects/wsbpel-comp-repo/wiki/LoanApprovalDoc>
7. Zhou, Z.Q., Huang, D.H., Tse, T.H., Yang, Z., Huang, H., Chen, T.Y.: Metamorphic testing and its applications. In: Proceedings of the 8th International Symposium on Future Software Technology (ISFST 2004). Software Engineers Association (2004)



# Opening up Context-aware Services Compositions to End-Users\*

Ignacio Mansanet<sup>1</sup>, Victoria Torres<sup>1</sup>, Pedro Valderas<sup>1</sup>, José Javier Berrocal<sup>2</sup>,  
José Manuel García<sup>2</sup>, Juan Manuel Murillo<sup>2</sup>, and Carlos Canal<sup>3</sup>

<sup>1</sup> Universitat Politècnica de València, Spain  
Pros Research Center

[imansanet](mailto:imansanet@pros.upv.es), [vtorres](mailto:vtorres@pros.upv.es), [pvalderas](mailto:pvalderas@pros.upv.es)

<sup>2</sup> Universidad de Extremadura, Spain

[jberolm](mailto:jberolm@unex.es), [jgaralo](mailto:jgaralo@unex.es), [juanmamu](mailto:juanmamu@unex.es)

<sup>3</sup> Universidad de Málaga, Spain  
[canal@cc.uma.es](mailto:canal@cc.uma.es)

**Abstract.** The interconnected world in which we live opens many possibilities to create, consume, and share knowledge and services. Even though end-users are more than ever prepared in terms of technology (e.g., by using smartphones), their specific context (i.e., personal interests, geographical location, etc.) is not yet properly considered in existing solutions to explore these possibilities. Therefore, we need to provide end-users with tools that allow them to create, consume, and share added value services by using the proper knowledge and services according to their context. In this sense this paper discusses how existing solutions could be integrated to achieve this goal. In particular we explore the possibility of extending EUCalipTool, an end-user mobile tool for service compositions, with the context-aware notification capabilities offered by nimBees.

**Keywords:** Service Discovery, Context-aware discovery, End-user Development

## 1 Introduction

Nowadays we live surrounded by an ecosystem of devices, applications, services, and data that allows monitoring and supporting our daily activities (e.g., commuting from home to work or monitoring users health (sport activities, blood pressure, sleeping quantity and quality, ingested calories, etc.)). Within this ecosystem, the amount of services that are made available is growing exponentially, turning their search into a cumbersome task. In addition, many times

---

\* This work has been developed with the support of MINECO under the SMART-ADAPT TIN2013-42981-P project, the RCIS TIN2014-53986-REDT network, the TIN2015-67083-R project, the TIN2015-69957-R project and co-funded by the ERDF, the Consejería de Economía e Infraestructura of the Junta de Extremadura (GR15098) and by FEDER funds.

users would like to make a combined usage of these services in order to better fulfill their specific needs. However, there are not yet tools targeted to end-users that allow them managing available services effectively.

Within this context, to achieve a more effective usage of all these services a new type of tools is required. In order to be successful, this type of tools should 1) take into account the profile of the user (e.g., age, gender, etc.) as well as her current context (e.g., location, season time, etc.), 2) run in mobile devices (e.g. smart phones), and 3) provide a user-friendly interfaces that stimulate users to use them.

As a first approach in this work we explore the possibility of combining the capabilities offered by two existing tools which are EUCalipTool[1,2] and nimBees<sup>4</sup>. On the one hand, EUCalipTool is a mobile tool that supports end-users during the composition of services from an integrated repository. Despite the fact that services and applications registered in EUCalipTool are tagged semantically, currently, it is the end-user who has to perform the search and selection of the interesting ones at each moment. On the other hand, nimBees is a push notification platform[3] based on the the People as a Service (PeaaS) paradigm[4] that uses the capabilities of modern mobile devices to gather the sociological profiles of their owners and provide them as services tightly related to their owners context.

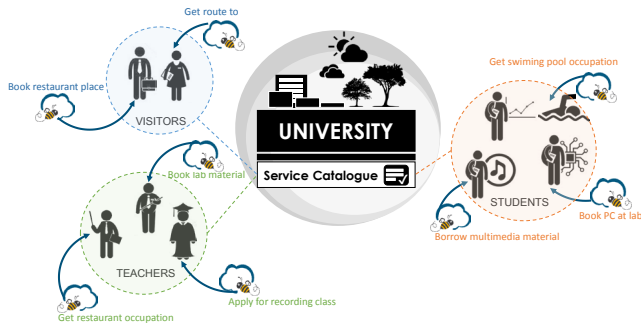


Fig. 1. Services are suggested based on the user profile and context

Therefore, by integrating these two tools we are endowing EUCalipTool with context-aware capabilities. This will help end-users finding services according to their current needs, i.e., based on their profile, preferences and location.

For illustration purposes take a look at figure 1. This figure shows a possible scenario at the university where different type of users exist (e.g., students,

<sup>4</sup> <http://www.nimbees.com/>



teachers and visitors). The university manages a services catalogue (or repository) with all the services offered to the community. Based on the user profile, preferences and context users are suggested about potential useful services at this exact moment. These suggestions can then be taken by the user to be part of a new composition.

## 2 Integration Plan

To achieve the integration between EUCalipTool and nimBees requires reconsidering some of the architectural aspects initially designed in EUCalipTool. In particular these aspects refer to the EUCalipTool repository and its service search and sharing mechanisms.

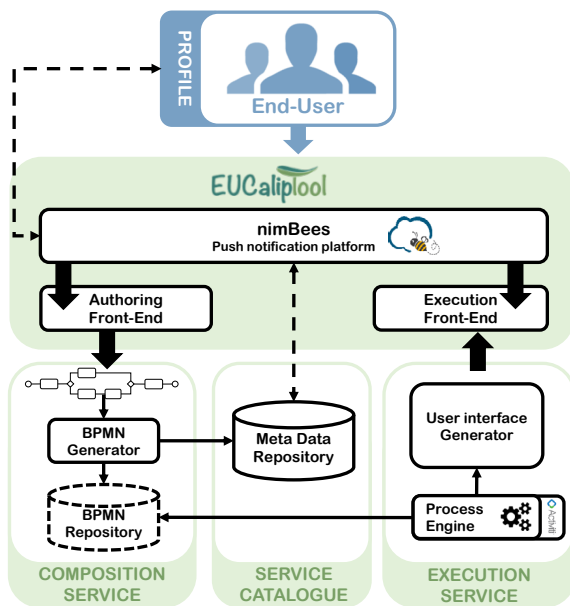


Fig. 2. nimBees integration into the EUCalipTool architecture

As figure 2 shows, nimBees should be able to access and search the services registered in the catalogue. For this purpose, services are tagged semantically,

so these tags can be used to match the user preferences, profile and context. The result of this search is then sent to end-users via the corresponding Front-end so they can use potential interesting services either to directly execute them (via the Execution front-end) or to use them as part of a new composition (via the Authoring Front-End). All the knowledge produced during this process (e.g., the individual/composed usage of service  $S$  in context  $C$  by user  $U$ ) is stored in the repository as valuable data to improve the service selection.

Concerning the sharing mechanisms, the push notification mechanism offered by nimBees allows reaching not just users connected through social networks as was initially designed in EUCalipTool. Instead, potentially interesting services are sent to users that show common interests (through their profiles) and that are reachable at that precise moment.

### 3 Expected Benefits

Up to date, services and applications are designed to be used by end-users as these were conceived originally. However, it has been proven that the one-size-fits-all approach is no longer valid and that new tools, approaches, and mechanisms are required to better fit users needs. Therefore, by integrating EUCalipTool and nimBees we expect to get closer existing services to end-users based on their current needs. By using nimBees notification capabilities, service compositions can be shared with other users that may be interested on them based on their profile (i.e., their preferences) and also their current context (e.g., their current location).

### References

1. Mansanet, I., Torres, V., Valderas, P., Pelechano, V.: A Mobile End-User Tool for Service Compositions. In: Actas de las X Jornadas de Ciencia e Ingeniería de Servicios (JCIS). (2014) 25–35
2. Mansanet, I., Torres, V., Valderas, P., Pelechano, V.: A Mobile End-User Tool for Service Compositions. In: Biblioteca Digital de Sistedes. Actas de las XI Jornadas de Ciencia e Ingeniería de Servicios (JCIS). (2015)
3. Miranda, J., Makitalo, N., Garcia-Alonso, J., Berrocal, J., Mikkonen, T., Canal, C., Murillo, J.: From the internet of things to the internet of people. *Internet Computing, IEEE* **19** (2015) 40–47
4. Guillen, J., Miranda, J., Berrocal, J., Garcia-Alonso, J., Murillo, J.M., Canal, C.: People as a Service: A mobile-centric model for providing collective sociological profiles. *IEEE Software* **31** (2014) 48–53

# Construyendo Perfiles Virtuales Mediante el Procesamiento de Eventos Complejos

Javier Berrocal<sup>1</sup>, Juan Boubeta-Puig<sup>2</sup>, Carlos Canal<sup>3</sup>, Jose Garcia-Alonso<sup>1</sup>,  
Juan M. Murillo<sup>1</sup>, and Guadalupe Ortiz<sup>2</sup>

<sup>1</sup> Dpto. de Ingeniería de Sistemas Informáticos y Telemáticos, Universidad de Extremadura, España

{jberolm,jgaralo,juanmamu}@unex.es

<sup>2</sup> Dpto. de Ingeniería Informática, Universidad de Cádiz, España

{juan.boubeta,guadalupe.ortiz}@uca.es

<sup>3</sup> Dpto. de Lenguajes y Ciencias de la Computación, Universidad de Málaga, España  
canal@lcc.uma.es

**Resumen** A medida que se incrementa el número de dispositivos inteligentes, el esfuerzo requerido para adaptarlos a las necesidades de cada usuario también crece. Asimismo, el proceso de adaptación de un dispositivo al contexto de un usuario es todavía un proceso muy manual. A pesar de que en los últimos años han surgido algunas propuestas centradas en obtener la información contextual de los usuarios para crear sus perfiles virtuales, se necesitan soluciones novedosas que permitan crear perfiles más completos, que sean utilizados por los dispositivos inteligentes para adaptarse automáticamente a las necesidades de sus usuarios, redundando en una mayor exactitud de la adaptación. En este artículo se propone la integración del modelo computacional *People as a Service* (PeaaS) con el procesamiento de eventos complejos (CEP) para la creación en tiempo real de perfiles virtuales complejos desde el propio dispositivo móvil y la compartición de estos como servicios para el resto de sistemas y dispositivos. Además, se evalúa esta integración en un caso de estudio sobre Alzheimer. Los resultados confirman que el uso de la tecnología CEP para la identificación de información contextual compleja es posible.

**Keywords:** Información Contextual, CEP, PeaaS, Asper, MDD

## 1. Introducción

El incremento en las capacidades de los dispositivos embebidos ha permitido el desarrollo de *Dispositivos Inteligentes*. Estos dispositivos normalmente están conectados a Internet y proporcionan una representación virtual de ellos en la red contra la que otros dispositivos pueden interactuar. Esta red de dispositivos inteligentes interconectados ha permitido el desarrollo de la Internet de las Cosas (*Internet of Things*, IoT) [13] y la Web de las Cosas (*Web of Things*, WoT) [15]. Uno de los principales objetivos de estos paradigmas es simplificar la vida de las personas haciendo que la tecnología trabaje para ellos.

Se estima que en el año 2020 habrá entre 50 y 100 billones de dispositivos inteligentes conectados a Internet [18]. Actualmente, con el objetivo de aumentar la usabilidad, el comportamiento de estos dispositivos y de los sistemas basados en WoT pueden adaptarse a las preferencias y al contexto del usuario, el cual puede además cambiar considerablemente a lo largo del tiempo.

Sin embargo, el proceso de identificación y adaptación de un sistema o dispositivo al contexto de un usuario es todavía un proceso muy manual. Debe ser el usuario el que establezca las nuevas preferencias y el nuevo contexto para cada dispositivo y sistema. Tener que configurar en el futuro todos los dispositivos que darán soporte a las actividades cotidianas requerirá un gran esfuerzo, que tendrá que ser duplicado cada vez que haya un cambio en el contexto. Esta configuración manual es aceptable cuando se interactúa con un conjunto pequeño de dispositivos; sin embargo, si este número se incrementa como está previsto, la carga de gestión de los dispositivos requerirá un esfuerzo que reducirá notablemente los beneficios proporcionados por estos sistemas.

Por lo tanto, se necesitan soluciones que permitan identificar el contexto y las preferencias de las personas, para que posteriormente sea aplicado sobre los dispositivos. Actualmente, existen varios trabajos que definen distintas propuestas para crear el perfil virtual de una persona [2], [17]. De hecho, los autores de este artículo incluso han propuesto un modelo computacional móvil, llamado *People as a Service* (PeaaS) [14], que convierte al teléfono móvil en el elemento central para capturar y mantener dicho perfil. Este perfil es posteriormente provisto como un servicio para que el resto de sistemas y dispositivos puedan obtener la información necesaria para adaptarse a las necesidades del usuario. Además, para asegurar la privacidad de los datos, uno de los pilares de PeaaS es que cada usuario pueda definir qué sistemas externos pueden acceder a la información contextual.

Estos perfiles normalmente contienen información de bajo nivel obtenida de distintas fuentes y sensores. Con el propósito de poder identificar información contextual de alto nivel y, así, crear perfiles virtuales más completos, en este artículo se propone la integración de PeaaS con el procesamiento de eventos complejos (*Complex Event Processing*, CEP). CEP es una tecnología que permite analizar ingentes volúmenes de datos para detectar situaciones de interés en tiempo real, gracias al uso de un software específico: un motor CEP. Concretamente, este motor CEP recibirá como entrada eventos sobre los cambios contextuales de una persona. Entonces, estos eventos serán analizados junto con la información histórica y, en base a determinados patrones de eventos predefinidos, será identificada la información de alto nivel.

El resto del artículo se estructura de la siguiente manera. La Sección 2 propone una arquitectura novedosa para integrar PeaaS con un motor CEP, y especifica cómo se hace uso de CEP para la obtención de información contextual compleja. La Sección 3 describe un caso de estudio sobre Alzheimer en el que se aplica dicha arquitectura al ámbito de la salud. La Sección 4 especifica los trabajos relacionados. Finalmente, la Sección 5 presenta las conclusiones y el trabajo futuro.

## 2. Integración PeaaS y CEP

En los últimos años estamos viendo un considerable incremento en las capacidades de almacenamiento y cómputo de los teléfonos móviles. Las últimas investigaciones demuestran que es posible utilizarlos para realizar tareas que anteriormente estaban relegadas a un PC o un servidor (como construir y almacenar el perfil virtual de un usuario [14] o la ejecución de motores CEP [12]). En esta sección se detalla la integración de un motor CEP en el modelo computacional PeaaS con el objetivo de poder crear perfiles virtuales más completos. Además, siguiendo la filosofía PeaaS, tanto el perfil como el motor se encontrarían en el dispositivo final, facilitando la escalabilidad del sistema. De esta forma, las aplicaciones instaladas en los dispositivos móviles podrán hacer uso de dichos perfiles para adaptar su comportamiento, y el comportamiento de otros dispositivos asociadas a ellas.

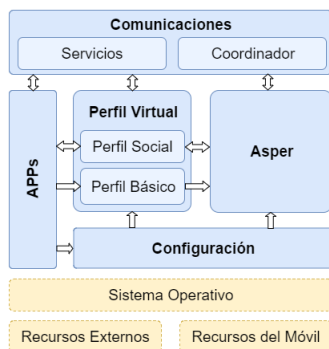


Figura 1. Arquitectura de PeaaS integrando el motor CEP.

La Figura 1 muestra una posible arquitectura de alto nivel de esta integración. Para mantener la filosofía de PeaaS, todos los módulos y responsabilidades de la arquitectura recaen sobre el dispositivo móvil. Así, el dispositivo móvil adquiere el rol principal en la arquitectura y se garantiza que el perfil virtual se encuentre únicamente almacenado en el dispositivo del usuario. Esto no significa que no se puedan utilizar otros sistemas externos, pero tienen un rol secundario en la arquitectura.

Debido a las restricciones de los sistemas operativos, la integración PeaaS-CEP está concebida para que sea implementada a nivel de aplicación. Por ello, todos sus módulos recaen sobre el sistema operativo para acceder a funcionalidades específicas y a los recursos del móvil o externos. Los *Recursos del Móvil* son

los distintos sensores del dispositivo (i.e., GPS, acelerómetro, sensor de luminosidad, etc.). Los *Recursos Externos* son sensores no incorporados en el dispositivo móvil pero con los que está conectado (pulseras cuantificadoras, sensores de pulso, beacons, etc.).

El elemento central de la arquitectura es el *Perfil Virtual*. Este módulo es el encargado de obtener y almacenar la información contextual del usuario. En concreto, este perfil puede obtener información de los recursos internos o externos conectados al móvil, de las aplicaciones que hacen uso de la arquitectura –por ejemplo, sobre cómo el usuario interactúa con ellas– o de servicios externos –como servicios meteorológicos, de tráfico, etc. Una particularidad de este módulo es que no sólo almacena la última y más actualizada información contextual, sino que mantiene un histórico con toda la información recopilada. Esta información es almacenada siguiendo un orden temporal, i.e. formando un *timeline*, de forma que también se pueden hacer inferencias y consultas basadas en la evolución y los cambios de un determinado dato en el tiempo. El perfil virtual es dividido en el *Perfil Básico* y el *Perfil Social*. El Perfil Básico contiene todos los datos en bruto obtenidos de los distintos recursos y servicios. El Perfil Social contiene el resultado del procesamiento de los datos en bruto y los eventos complejos obtenidos por el motor CEP.

Asper [11] es el motor CEP móvil propuesto para esta arquitectura. Este motor, una adaptación del motor CEP Esper para Android, recibe información del perfil virtual y del módulo de comunicaciones. De los perfiles básico y social obtiene los datos capturados de los distintos recursos y la información de alto nivel detectada con anterioridad. Del módulo de comunicaciones puede recibir información o eventos derivados de la invocación de distintos servicios desplegados en el móvil como, por ejemplo, la recepción de una notificación *push* con un determinado contenido en el mensaje. Toda la información obtenida es procesada por el motor para, primero, identificar información contextual de alto nivel –como cuál es la residencia habitual de una persona o su temperatura de confort– y, segundo, para la detección de si se cumplen determinadas condiciones que conlleven la ejecución de acciones –por ejemplo, si se detecta que un usuario llegará tarde a casa, se podría indicar al termostato que controla la temperatura de la vivienda del usuario para que retrase su encendido. Cuando el motor detecta información contextual de alto nivel ésta es almacenada en el perfil social, para que pueda ser accedida por cualquier aplicación. Cuando el motor detecta un patrón de eventos relativo a una acción que debe ser realizada, ésta es notificada a la aplicación, para que se dispare alguna funcionalidad o comportamiento, y/o al submódulo coordinador, en el caso de que sea necesario coordinar varias personas y dispositivos externos para completar la funcionalidad.

El módulo de *Comunicaciones* es el encargado de gestionar las interacciones con servicios y dispositivos externos no conectados directamente con el dispositivo móvil. Este módulo está dividido en dos submódulos: *Servicios* y *Coordinador*. El submódulo Servicios gestiona la comunicación con servicios externos. Este submódulo se relaciona con las aplicaciones basadas en esta arquitectura, proporcionándoles servicios de envío y recepción de notificaciones *push* basadas

en el contenido del perfil virtual; con el perfil virtual, para añadir información invocando servicios web; y con el motor CEP, para la redirección de notificaciones *push* requeridas para la identificación de patrones de eventos o para la invocación de determinados servicios como resultado de la ejecución de una acción tras la detección de un patrón. El módulo Coordinador es el encargado de coordinar distintos dispositivos cuando el motor CEP o una aplicación detecta que se debe realizar una acción en la que todos ellos están involucrados. Por ejemplo, si el motor CEP detecta que una persona tiene invitados en casa, se podría enviar un mensaje a todos ellos para saber cuál es su temperatura de confort y, así, posteriormente poder establecer en el termostato la temperatura adecuada.

El módulo de *Configuración* contiene información sobre los recursos y servicios de los que el perfil virtual tiene que obtener información, la frecuencia mínima con la que se debería obtener la información, y los patrones de eventos que tendrá que procesar el motor CEP. Esta información es proporcionada inicialmente al módulo de Configuración por cada una de las aplicaciones que hacen uso del perfil virtual, pero también pueden cambiar en tiempo de ejecución mediante la recepción de notificaciones *push* con dicha configuración. Así, por ejemplo, si en tiempo de ejecución se desea modificar los patrones de eventos, el experto del dominio podría utilizar cualquier ordenador para definir gráficamente nuevos patrones, o modificar los existentes, utilizando MEdit4CEP [6]. Una vez definidos, dichos patrones pueden ser incorporados a una notificación *push* y enviada al dispositivo de los usuarios de los que se desea adaptar el comportamiento. Esta notificación es recibida y gestionada por el módulo de Comunicaciones para actualizar los datos almacenados en el módulo de Configuración. El mismo proceso es utilizado para actualizar la frecuencia de obtención de información de los distintos recursos. El módulo de Configuración almacena y unifica la información de acceso a los sensores y los patrones de eventos.

Finalmente, el módulo *APPs* conforma las distintas aplicaciones que hacen uso del perfil virtual.

### 3. Caso de Estudio: SafeWalks y Asper

En esta sección se detalla un caso de estudio del uso de la integración PaaS-CEP para la construcción de perfiles virtuales. SafeWalks [16] es una aplicación que monitoriza los movimientos de personas con una capacidad cognitiva reducida o en un estadio temprano de Alzheimer, pero que aún tienen cierta autonomía y viven en sus casas.

Esta aplicación monitoriza a estas personas creando un perfil virtual lo más completo posible. Esta información es posteriormente procesada para identificar las rutinas de una persona –i.e. a qué hora se levanta, a qué hora suele salir de paseo, cuál es el recorrido que suele realizar, cómo influyen las condiciones meteorológicas en dichas actividades, etc.– o si hay desviaciones de rutinas, es decir, que un usuario no esté realizando la tarea esperada, lo que podría indicar que está perdido. Cuando se identifican desviaciones se disparan distintas acciones para tratarlas; por ejemplo, notificando las desviaciones a los cuidadores

asociados con el usuario. Mediante la identificación de rutinas y desviaciones se consigue mejorar la seguridad y calidad de vida de los usuarios y de los cuidadores.

Esta aplicación se basa en el modelo computacional PeaaS. Para ello, hace uso de la implementación comercial nimBees [1]. Esta implementación cubre los módulos de Comunicaciones y Perfil Virtual de nuestra arquitectura descrita en la Sección 2. Sin embargo, en una primera versión de la aplicación, la detección de rutinas y de desviaciones fue directamente implementada mediante código Java. Así pues, en este caso de estudio se aborda la integración del motor CEP Asper en dicha aplicación y la migración de parte del código de detección de rutinas y desviaciones, con el objetivo de evaluar si es factible la utilización de estos motores para la identificación de información contextual de alto nivel.

La información contextual más importante captada por esta aplicación es: localización en el interior de la vivienda del usuario –para lo que se hace uso de Beacons [3]–, localización en exteriores –mediante el sensor de GPS del móvil–, las personas que acompañan al usuario –para lo que se utilizan las capacidades de comunicación de nimBees– y el clima –que se obtiene invocando a un servicio externo. Toda la información capturada es almacenada en un histórico ordenado según la hora del día, lo que permite identificar las rutinas diarias del usuario.

Las acciones más importantes del flujo de información para la construcción del perfil virtual del paciente son las siguientes: (1) el Perfil Virtual obtiene información de los sensores según la frecuencia definida por la configuración de la aplicación, (2) la información recuperada es almacenada en el Perfil Básico y enviada al motor CEP, (3) el motor CEP ejecuta los patrones definidos en la configuración para detectar eventos complejos, (4) si se detecta información de alto nivel, ésta es almacenada en el Perfil Social –por ejemplo, si se detecta una rutina que se está realizando, ésta se reforzaría– y (5) si se detecta una desviación de la rutina, ésta es notificada a los cuidadores mediante el uso de la plataforma nimBees.

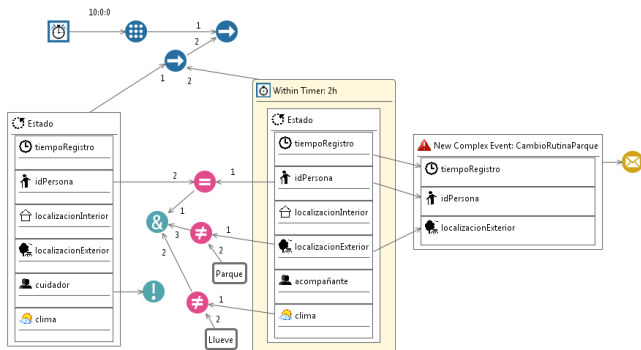
Para que el motor CEP pueda detectar información de alto nivel o desviaciones de rutinas, se han definido una serie de patrones de eventos; por falta de espacio sólo se presentan dos a modo de ejemplo:

- **CambioRutinaParque:** este patrón comprueba cada día a las 10.00 h. si el usuario con Alzheimer que se encuentra solo en ese momento (sin cuidador) y, a pesar de que no está lloviendo, no está en el parque durante las próximas 2 horas. En ese caso, se enviará un correo electrónico alertando de esta situación tanto a su cuidador como a sus familiares. La Figura 2 muestra este patrón modelado gráficamente con la herramienta MEdit4CEP [6].
- **CambioRutinaBiblioteca:** este patrón comprueba cada día a las 10.00 h. si el usuario sin cuidador y, aunque está lloviendo, no está en la biblioteca durante las próximas 2 horas.

Cabe destacar que el uso de la herramienta MEdit4CEP para definir los patrones de eventos nos proporciona varias ventajas, entre las que destacan: (1) los expertos en Alzheimer pueden definir por sí mismos –sin ayuda de informáticos



ni expertos en CEP [5]– tanto las condiciones que deben cumplirse para detectar cambios de rutinas de los pacientes como las acciones que deben llevarse a cabo tras su detección, y (2) los modelos gráficos de estos patrones pueden transformarse automáticamente al código –correcto y libre de errores– desplegable en los motores CEP, gracias a las técnicas de desarrollo dirigido por modelos empleadas. Por lo tanto, si un experto en Alzheimer desea modificar algún patrón, para mejorar su precisión o adaptarlo a un comportamiento concreto, puede utilizar MEdit4CEP para definir gráficamente el nuevo patrón y transformarlo automáticamente a código, y la plataforma Web de nimBees para enviar notificaciones *push* a las aplicaciones de los pacientes sobre las que se desea aplicar los nuevos patrones.



**Figura 2.** Patrón de eventos *CambioRutinaParque* modelado con MEdit4CEP.

Las pruebas iniciales realizadas sobre este caso de estudio nos han permitido comprobar que la utilización de motores CEP es una solución viable para la identificación tanto de información contextual compleja como de las acciones concretas que deben ejecutarse en cada momento. La utilización de estos motores permite incluir o modificar fácilmente los patrones de eventos para la detección en tiempo real de dicha información compleja. No obstante, el uso de un motor CEP en el dispositivo móvil podría conllevar un consumo de recursos más elevado que la implementación de las reglas de forma *ad-hoc* usando los lenguajes de programación específicos de cada dispositivo. En trabajos futuros se comparará este rendimiento.

## 4. Trabajos Relacionados

Desde el surgimiento de la tecnología CEP, los motores CEP se han ejecutado normalmente en ordenadores de escritorio o bien en servidores más potentes con el propósito de detectar situaciones críticas o relevantes en tiempo real para un dominio en particular. Con el avance de la tecnología, los dispositivos móviles son cada vez más potentes, lo que posibilita la aplicación de CEP en estos dispositivos.

Dunkel et al. [10] han denominado *mobile CEP* a este enfoque emergente en el que se hace uso de la tecnología CEP en *smartphones*. Según los autores, este enfoque, adoptado en nuestro trabajo, dispone de dos características principales: (1) conciencia de la situación, mediante la correlación de datos provenientes de los distintos tipos de sensores localizados en los propios teléfonos inteligentes y (2) conciencia del contexto, a través de la incorporación tanto de datos del contexto, como de dispositivos periféricos. Como ventajas significativas de este enfoque, destacan las siguientes: (1) procesamiento de los datos de los sensores sin necesidad de establecer ninguna comunicación con el servidor, privacidad de la información y correlación de datos en el propio dispositivo. En este trabajo, aplican este enfoque a un escenario de vida cotidiana asistida por el entorno (*Ambient Assisted Living*, AAL), concretamente implementan una app Android para la detección automática de caídas de personas ancianas. Mientras los autores definen los patrones de eventos mediante las *condiciones* que deben cumplirse para detectar una situación y el tipo de evento complejo (denominado *action*) que debe crearse tras su detección, en nuestro trabajo definimos los patrones de eventos como *condiciones*, *tipo de evento complejo* y *acciones* conforme a nuestros lenguajes específicos de dominio ModeL4CEP [7]. De esta forma, separamos la definición del propio patrón de las posibles acciones que deberían llevarse a cabo tras su detección como, por ejemplo, el envío de un correo electrónico o la invocación de un servicio.

Bruns et al. [8] también proponen el uso de CEP en el móvil para coordinar la gestión eficiente del personal de ambulancias ante un accidente de tráfico. Como novedad, utilizan además una máquina de estados finitos (*Finite State Machine*, FSM) para representar los estados de operación de una ambulancia. Gracias a la integración del motor CEP Asper [11] con FSM, se pueden detectar automáticamente cambios de dichos estados de operación. Este trabajo presenta la misma limitación del anterior en cuanto a la definición de los patrones de eventos.

Birth et al. [4] describen una EDA integrada con CEP y también implementan una app Android, haciendo uso de Asper, para proporcionar a los viajeros información en tiempo real sobre situaciones inesperadas que ocurran durante la ruta seleccionada por el usuario. La principal limitación de este trabajo es la dependencia de la arquitectura propuesta a este dominio de aplicación.

Finalmente, Coronado et al. [9] definen un enfoque basado en CEP distribuido para coordinar los servicios de automatización de tareas (*Task Automation Services*, TAS). Por un lado, los autores definen el concepto de “perfil” como el lugar (dispositivo móvil o servidor) en el que deben ejecutarse los patrones

de eventos. Estos perfiles se clasifican en 3 tipos: *web-driven execution* (ejecución de patrones en el servidor), *device-driven execution* (en el dispositivo) o *mixed execution* (distribución de perfiles entre el servidor y el dispositivo). Con respecto a la definición de patrones, como ya se ha mencionado, nosotros proporcionamos una herramienta de modelado gráfica (MEdit4CEP) que permite a los expertos en el dominio la definición de los mismos. Estos autores también implementan una app Android integrada con el motor Asper. Sin embargo, el servidor de coordinación envía (*push*) las notificaciones a los *smartphones* usando *Google Cloud Messaging* (GCM), mientras en nuestro trabajo se lleva a cabo con la plataforma nimBees, una solución que se ajusta mejor a las necesidades de los sistemas adaptados al contexto, ya que permite enviar notificaciones que pueden tener asociadas condiciones basadas en el contenido del perfil virtual del usuario.

## 5. Conclusiones y Trabajo Futuro

El aumento de las capacidades de los teléfonos móviles permite crear aplicaciones que puedan realizar funcionalidades cada vez más complejas. Actualmente, los usuarios están demandando que dichas funcionalidades sean capaces de cubrir sus necesidades con la menor configuración posible. La consecución de este objetivo será crucial para el éxito de muchas aplicaciones. En este artículo se integra el modelo computacional PeaaS con un motor CEP con la finalidad de crear perfiles virtuales más completos y de facilitar la identificación de situaciones contextuales complejas que conllevan a las aplicaciones a la ejecución de acciones.

Puesto que el consumo de recursos, y especialmente la batería, es especialmente sensible para las aplicaciones y los teléfonos móviles, actualmente estamos trabajando en evaluar el impacto que conlleva el uso de motores CEP para la identificación de información contextual compleja.

**Agradecimientos.** Este trabajo ha sido parcialmente financiado por el Ministerio de Economía y Competitividad (proyectos TIN2014-53986-REDT, TIN2014-53555-REDT, TIN2015-67083-R, TIN2015-69957-R y TIN2015-65845-C3-3-R), la Junta de Extremadura (GR15098) y los fondos FEDER. Boubeta-Puig agradece la hospitalidad recibida durante su estancia de investigación con el Grupo Quercus de Ingeniería del Software de la Universidad de Extremadura, donde parte de este trabajo ha sido desarrollado.

## Referencias

1. nimBees. <http://www.nimbees.com>, [Online]
2. Abowd, G.D., Dey, A.K., Brown, P.J., Davies, N., Smith, M., Steggles, P.: Towards a better understanding of context and context-awareness. In: Int. Symposium on Handheld and Ubiquitous Computing. pp. 304-307. Springer-Verlag (1999)

3. Apple Inc: ibeacon for developers. <https://developer.apple.com/ibeacon/>, [Online]
4. Birth, O., Hoffmann, M., Strassberger, M., Roor, R., Schlichter, J.: Concept for an Intermodal Traveller Information System with Real-Time Data Using Complex Event Processing. In: 2015 IEEE 18th International Conference on Intelligent Transportation Systems (ITSC). pp. 2293–2298 (Sep 2015)
5. Boubeta-Puig, J.: Desarrollo Dirigido por Modelos de Interfaces Específicas de Dominio para el Procesamiento de Eventos Complejos en Arquitecturas Orientadas a Servicios. Ph.D. thesis, Universidad de Cádiz, Cádiz, España (Jul 2014), <http://rodin.uca.es:80/xmlui/handle/10498/17554>
6. Boubeta-Puig, J., Ortiz, G., Medina-Bulo, I.: MEdit4CEP: A model-driven solution for real-time decision making in SOA 2.0. Knowledge-Based Systems 89, 97–112 (Nov 2015)
7. Boubeta-Puig, J., Ortiz, G., Medina-Bulo, I.: ModeL4CEP: Graphical domain-specific modeling languages for CEP domains and event patterns. Expert Systems with Applications 42(21), 8095–8110 (Nov 2015)
8. Bruns, R., Dunkel, J., Billhardt, H., Lujak, M., Ossowski, S.: Using Complex Event Processing to support data fusion for ambulance coordination. In: 2014 17th International Conference on Information Fusion (FUSION). pp. 1–7 (Jul 2014)
9. Coronado, M., Bruns, R., Dunkel, J., Stipkovic, S.: Context-Awareness in Task Automation Services by Distributed Event Processing. In: Benatallah, B., Bestavros, A., Catania, B., Haller, A., Manolopoulos, Y., Vakali, A., Zhang, Y. (eds.) Web Information Systems Engineering – WISE 2014 Workshops, Lecture Notes in Computer Science, vol. 9051, pp. 190–203. Springer International Publishing, Switzerland (2015)
10. Dunkel, J., Bruns, R., Stipkovic, S.: Event-based smartphone sensor processing for ambient assisted living. In: 2013 IEEE Eleventh International Symposium on Autonomous Decentralized Systems (ISADS). pp. 1–6 (Mar 2013)
11. Eggum, M.: Asper - Esper for Android (2014), <https://github.com/mobile-event-processing/Asper>, [Online]
12. Eggum, M.: Smartphone Assisted, Complex Event Processing. Enablement of Esper on Android-based devices (2014), <https://www.duo.uio.no/bitstream/handle/10852/41663/Marcel-Eggum---Thesis.pdf?sequence=1&isAllowed=y>, Master Thesis [Online]
13. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of Things (IoT): A vision, architectural elements, and future directions. Future Generation Computer Systems 29(7), 1645 – 1660 (2013)
14. Guillen, J., Miranda, J., Berrocal, J., Garcia-Alonso, J., Murillo, J.M., Canal, C.: People as a Service: A mobile-centric model for providing collective sociological profiles. IEEE Software 31(2), 48–53 (2014)
15. Guinard, D.: Towards the web of things: Web mashups for embedded devices. In: In MEM 2009 in Proceedings of WWW 2009. ACM (2009)
16. Lozano, P.P., Vereda, A.P., Murillo, J.M., Canal, C.: Safewalks: aplicación móvil de supervisión de pacientes de alzheimer. In: Jornadas de Ciencia e Ingeniería de Servicios (JCIS) (2015)
17. Makris, P., Skoutas, D.N., Skianis, C.: A survey on context-aware mobile and wireless networking: On networking and computing environments’ integration. IEEE Communications Surveys and Tutorials 15(1), 362–386 (2013)
18. Perera, C., Liu, C.H., Jayawardena, S., Chen, M.: Context-aware computing in the internet of things: A survey on internet of things from industrial market perspective. CoRR (2015)

# La Web de las Cosas se nos viene encima

Carlos Canal<sup>1</sup>, Juan Manuel Murillo<sup>2</sup>

<sup>1</sup> Universidad de Málaga  
canal@lcc.uma.es

<sup>2</sup> Universidad de Extremadura  
juanmamu@unex.es

**Resumen.** A medida que crece el interés por la denominada Web de las Cosas (WoT), deberíamos hacer disminuir las barreras de entrada para el uso de las tecnologías asociadas. Hoy en día somos capaces de desarrollar aplicaciones que adaptan su comportamiento de acuerdo a condiciones definidas de antemano, así como a las preferencias personales de sus usuarios, facilitando así su utilización. El software para la Web de las Cosas que desarrollemos en el futuro inmediato debería ser capaz de ajustar de forma automática su comportamiento también de acuerdo a situaciones no predefinidas y al contexto en el que se mueven sus usuarios. En este artículo de reflexión, discutimos el estado actual del arte y la necesidad de nuevos modelos y herramientas capaces de hacer frente a estos retos, de forma que podamos predecir el comportamiento esperado de un sistema WoT y la interacción necesaria entre los dispositivos que lo integran, con el objetivo de lograr una mejor respuesta del sistema a información contextual variable.

**Palabras clave:** Internet de las Cosas, Web de las Cosas, Sensibilidad al contexto

El imparable crecimiento y evolución de Internet, desde sus inicios hasta la actualidad, ha ido pasando por distintos estadios, si atendemos al tipo de entidades que se comunican a través de la denominada red de redes. Si bien el objetivo inicial parecía consistir en un sistema que permitiese interconectar ordenadores y compartir datos, pronto se convirtió en un medio de comunicación entre las personas (e-mail) y de acceso de estas a la información (webs estáticas). El siguiente paso consistió en integrar en la red los sistemas software, permitiendo que interactuasen entre sí (servicios web) y también con las personas (webs dinámicas). En la actualidad, la virtualidad de la red está trascendiendo sus propias fronteras para dar el salto al mundo físico y convertirse en un sistema en el que también las cosas están integradas y pueden comunicarse entre sí y con las personas y el software.

En efecto, algunos estudios predicen para 2020 habrá entre cincuenta y cien mil millones de dispositivos de todo tipo conectados a Internet [12]. Entre ellos obviamente se contarán los ordenadores, incluyendo los personales, así como las tabletas y teléfonos móviles, pero también dispositivos inteligentes (relojes, gafas y otros *wearables*), otros grandes dispositivos con su propio punto de acceso a la red (automóviles, televisiones, ascensores), dispositivos no inteligentes, pero que al menos puedan ser monitorizados y controlados a través sistemas como *Arduino* o *Raspberry Pi* (entre ellos, todo tipo de electrodomésticos), y finalmente dispositivos inertes desde un punto de vista computacional, pero cuya identidad y localización puede ser controlada por ejemplo por medio de etiquetas RFID incorporadas (desde paquetes de mensajería a todo tipo de productos, pasando incluso por el ganado).

Esto es lo que se ha dado en llamar el Internet de las Cosas (IoT) [4], mediante

el cual los dispositivos conectados pueden ser localizados, accedidos y en última instancia controlados a través de Internet, lo que da lugar al desarrollo de aplicaciones para interactuar con estos dispositivos, ya sea de forma individual o combinada. Estas aplicaciones tienen unos propósitos que van por ejemplo desde el control centralizado de las viviendas inteligentes, hasta la gestión integral de los productos desarrollados por una empresa manufacturera, incluyendo la fabricación, almacenamiento, distribución y venta. No obstante, más que de Internet de las Cosas, deberíamos hablar de *intranets* aisladas [7], ya que dichas aplicaciones están pensadas para interactuar con una serie de dispositivos predeterminados y realizar con ellos funciones específicas.

Más recientemente se ha empezado a hablar de la Web de las Cosas (WoT) [6], en la que los dispositivos no son simplemente accesibles a través de Internet mediante protocolos propietarios específicos, sino que están plenamente integrados en la web, lo que permite la interacción mediante tecnologías estándar del nivel de aplicación, tales como HTTP, HTML, o JSON. Los dispositivos WoT proporcionan una representación virtual de sí mismos y de su interacción con otros dispositivos para formar sistemas complejos, dando lugar a nuevas aplicaciones y servicios en contextos mucho más abiertos y dinámicos que los habitualmente planteados en IoT.

La integración de miríadas de dispositivos en la WoT debería servir, entre otros objetivos, para simplificar la vida de las personas, facilitando la interacción con su entorno físico, ayudando en la realización de determinadas tareas o proporcionando más información para la toma de decisiones. Sin embargo, si analizamos cómo se realiza actualmente la interacción entre estos dispositivos y las personas, los beneficios no son los que en un principio cabría esperar. La razón es que el comportamiento de las cosas inteligentes y las aplicaciones relacionadas con ellos debe depender de las preferencias de sus usuarios, que pueden cambiar considerablemente a lo largo del tiempo, dependiendo de su contexto. Sin embargo, el modo en que estos sistemas se adaptan a ese contexto se lleva a cabo de una forma aún manual.

De este modo, la gestión de una colección de cosas inteligentes en continuo crecimiento y con las que interactuamos a diario va a requerir una atención constante. Además, cuando se produzcan cambios en el contexto o en las preferencias de los usuarios, será necesario reconfigurar los dispositivos. Como ejemplo, consideremos un termostato que permita monitorizar y ajustar la temperatura de una estancia. Para controlar este sistema el usuario establece un conjunto de temperaturas deseables para momentos específicos. Esta configuración debe ser modificada manualmente cuando cambian las preferencias del usuario o su contexto. Así, si hoy voy a llegar más temprano a casa puedo ajustar el sistema para hacer que la climatización comience a funcionar antes. Este modo manual de funcionamiento es aceptable cuando interactuamos con un número limitado de dispositivos, pero se convertirá en una carga inasumible cuando los usuarios tengan que gestionar docenas de sistemas conectados a la web. Debemos entonces plantearnos no solo qué nuevas aplicaciones y servicios pueden desarrollarse, sino también cómo va a ser la interacción entre las personas y la WoT, creando soluciones que integren de forma transparente y sin esfuerzo las preferencias y el contexto de las personas.

En la actualidad, ya hay grupos de investigación trabajando en cómo recopilar y procesar la información contextual de los usuarios con objeto de crear perfiles virtuales más completos [2], [10], [1], [13]. Entre ellos, los autores de este trabajo hemos propuesto el uso del teléfono móvil como el elemento clave para crear y gestionar estos perfiles [5], en lo que hemos dado en llamar *People as a Service* (PeaaS). PeaaS es un modelo de computación centrado en el móvil para inferir las preferencias y el contexto de su propietario y generar su perfil sociológico. Dicho perfil puede entonces utilizarse para orquestar de manera más efectiva los dispositivos inteligentes con los que interactúa el usuario.

No obstante, disponer de perfiles completos y precisos de los usuarios no es

suficiente. Se necesitan además técnicas para adaptar el comportamiento de las cosas a los cambios de contexto. El uso de técnicas de adaptación de software [9], composición dinámica [3], o programación orientada al contexto [8] permite establecer mecanismos básicos para definir el comportamiento de las aplicaciones dependiendo de información contextual específica. No obstante, estas técnicas asumen que la información y las variables que desencadenan las adaptaciones están detalladas en el código fuente de las aplicaciones, de modo que la adaptación se limita a un conjunto de situaciones preestablecidas. Sin embargo, no es factible prever durante la fase de desarrollo cada una de las situaciones posibles, especialmente en entornos dinámicos en los que las interacciones dependen del usuario pero también de su contexto, es decir, los dispositivos y las personas que están a su alrededor, su estado, su historia, etc. Volviendo a nuestro ejemplo, cuando vuelvo a casa, mi teléfono móvil puede interactuar con el termostato de la climatización para establecer la temperatura deseada, pero esta puede depender de si estoy solo o de quién me acompañe (y sus propias preferencias de temperatura de confort), de la actividad que esté realizando, o incluso, de mi estado de ánimo. De acuerdo con este planteamiento, nuestra propuesta de Internet of People [11] propone un manifiesto para sistemas WoT que admitan adaptaciones proactivas.

Por tanto, para poder gestionar la web de las cosas que se nos acerca, es necesario definir conceptos y modelos y desarrollar tecnologías que nos permitan adaptación en contextos altamente dinámicos. El objetivo es ser capaces de analizar las condiciones que se dan en un determinado tiempo y lugar, lo que podríamos denominar el *contexto de situación*, y usar este análisis para definir de manera dinámica el comportamiento que esperamos de los sistemas WoT. Este contexto situacional sería el resultado de combinar los perfiles virtuales de las diferentes entidades (tanto personas como dispositivos) involucradas en una determinada situación. En este contexto compuesto existirán entidades que definirán objetivos que detallan las condiciones que es deseable alcanzar, y entidades que proporcionen mecanismos capaces de hacer cumplir estos objetivos. De este modo, una vez que se compute el contexto de situación, deberá emerger del mismo el modo de actuación de las diferentes entidades para lograrlo. Esto requiere desarrollar una estrategia para alcanzar los objetivos conjuntos a partir de los mecanismos de los que disponen las entidades involucradas. Este contexto situacional proporcionará un mayor grado de integración entre los dispositivos inteligentes y las personas.

### Agradecimientos.

Este trabajo ha sido subvencionado parcialmente por el Ministerio de Ciencia e Innovación (proyectos TIN2014-53986-REDT, TIN2015-67083- R y TIN2015-69957-R), por el Departamento de Economía e Infraestructuras del Gobierno de Extremadura (GR15098), y por el fondo Europeo de Desarrollo Regional (FEDER).

### Bibliografía

1. Abowd, G.D., Dey, A.K., Brown, P.J., Davies, N., Smith, M., Steggle, P.: Towards a better understanding of context and context-awareness. In: Int. Symposium on Handheld and Ubiquitous Computing. pp. 304-307. Springer-Verlag (1999)
2. Bellavista, P., Corradi, A., Fanelli, M., Foschini, L.: A survey of context data distribution for mobile ubiquitous systems. ACM Comput. Surv. 44(4), 1-45 (2012)
3. Chen, G., Li, M., Kotz, D.: Data-centric middleware for context-aware pervasive computing. Pervasive Mob. Comput. 4(2), 216-253 (2008)
4. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of things (iot): A vision, architectural elements, and future directions. Future Generation Computer Systems 29(7), 1645-1660 (2013)
5. Guillen, J., Miranda, J., Berrocal, J., Garcia-Alonso, J., Murillo, J.M., Canal, C.: People as a service: A mobile-centric model for providing collective sociological profiles. IEEE Software 31(2), 48-53 (2014)

6. Guinard, D.: Towards the web of things: Web mashups for embedded devices. In: MEM 2009 in Proceedings of WWW 2009. ACM (2009)
7. Guinard, D., Trifa, V.M.: Building the Web of Things, Manning, 2016.
8. Hirschfeld, R., Costanza, P., Nierstrasz, O.: Context-oriented programming. Journal of Object Technology, March-April 2008, ETH Zurich 7(3), 125-151 (2008)
9. Kakousis, K., Paspallis, N., Papadopoulos, G.A.: A survey of software adaptation in mobile and ubiquitous computing. Enterprise Information Systems 4(4), 355-389 (2010)
10. Makris, P., Skoutas, D.N., Skianis, C.: A survey on context-aware mobile and wireless networking: On networking and computing environments' integration. IEEE Communications Surveys and Tutorials 15(1), 362-386 (2013)
11. Miranda, J., Makitalo, N., Garcia-Alonso, J., Berrocal, J., Mikkonen, T., Canal, C., Murillo, J.: From the internet of things to the internet of people. Internet Computing, IEEE 19(2), 40-47 (2015)
12. Perera, C., Liu, C.H., Jayawardena, S., Chen, M.: Context-aware computing in the internet of things: A survey on internet of things from industrial market perspective. CoRR (2015)
13. Raskino, J.F., Linden, A.: Extracting value from the massively connected world of 2015 (2015)



# Los servicios y la Nube



# Bidimensional Cross-Cloud Application Management with TOSCA and Brooklyn

Jose Carrasco, Javier Cubo, Francisco Durán, and Ernesto Pimentel

Universidad de Málaga, Dept. Lenguajes y Ciencias de la Computación, Spain  
{josec,cubo,duran,ernesto}@lcc.uma.es

**Abstract.** The diversity in the way different cloud providers offer their services, give their SLAs, present their QoS, support different technologies, etc., complicates the portability and interoperability of cloud applications, and favors vendor lock-in. Standards like TOSCA, and tools supporting them, have come to help in the provider-independent description of cloud applications. After the variety of proposed cross-cloud application management tools, we propose going one step further in the unification of cloud services with a deployment tool in which IaaS and PaaS services are integrated into a unified interface. We provide support for applications whose components are to be deployed on different providers, indistinctly using IaaS and PaaS services. The TOSCA standard is used to define a portable model describing the topology of the cloud applications and the required resources in an agnostic, and providers- and resources-independent way. We include in this paper some highlights on our implementation on Apache Brooklyn and present a non-trivial example that illustrates our approach.

**Keywords:** Cloud applications, multi-deployment, cross-cloud, standards, TOSCA, Brooklyn, IaaS, PaaS.

## *Trans-cloud* application deployment by unifying IaaS and PaaS

In recent years, Cloud Computing has experienced a growth in the demand of its services. The Cloud promotes access on demand to a large number of resources through out three service models, namely Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS), which allow cloud providers to offer a set of useful features for current IT requirements of the sector, among which we find scalability and elasticity. Given the interest on this computing model, vendors such as Google, Amazon, Cloud Foundry, etc. have implemented their solutions by developing their own cloud service layers with custom APIs that expose their resources. Most of these providers offer a set of similar services as regards functionality, but developed according to their own specifications. For example, each supplier specifies his own SLA or QoS, supports a concrete set of technologies, etc. The proliferation of these solutions has increased the number of issues to be addressed in cloud computing, mainly related to the diversity of providers and solutions, provoking the vendor lock-in problem, and hampering the portability and interoperability in cloud services.

In order to mitigate this heterogeneity and get a vendor-agnostic solution, independent tools and frameworks have emerged as the result of integrating, under a single interface, the services of multiple public and private providers. Over such interfaces, most of these solutions build models of application topologies, dependencies and used resources, independently of the providers in which services will be executed. Thus, they offer a portable and interoperable environment where developers can describe their systems and select the resources that better fit their requirements, without worrying about technical details of the services' use, and focusing on the required features.

In a very short time, these platforms have evolved according to the mode in which users can take advantage of integrated cloud services to expose and run their systems. Terms such as *multi-cloud*, *cross-cloud*, *federated clouds*, or *inter-clouds* have been used for deployment platforms with the ability of distributing modules of an application using services from different providers. The main differences between these terms are the different ways of handling the connections between modules deployed on different platforms. However, in all these attempts, platforms allow operating simultaneously with a single level of service to deploy applications, i.e., all the components of an application are deployed either at the IaaS level or all at the PaaS level. From this, with the goal of unifying cloud services, we propose a second dimension in which deployment tools integrate IaaS and PaaS levels under a single interface. Then, this will allow developers to deploy their applications combining services offered by providers at any of these levels. Following the evolution in terminology, *multi-/cross-/inter-cloud*, we envision *trans-cloud* management tools without the limitations we currently have. The idea behind *trans-cloud* is to be able to build our applications by using available services and resources offered by different providers, at IaaS, PaaS or SaaS level, using virtual machines or containers, according to our needs and preferences.

In this work, we present our proposal towards *trans-cloud* application deployment by unifying IaaS and PaaS of multiple vendors. Specifically, we propose using a provider-agnostic TOSCA-based model of the topology of applications and their required resources, indistinctly using IaaS and PaaS services, which can be used for their deployment using Apache Brooklyn.

Having an agnostic model of our system may greatly simplify migration, or simply decision change. Indeed, with our approach, each component may be deployed at one level or the other just by changing its location. Much has been said on choosing IaaS or PaaS for our applications. We may decide to go for PaaS when we want to provide flexibility or scalability to our applications without managing it ourselves. Or for IaaS, when we need control over our infrastructure, e.g., to accessing, monitoring or managing our virtual machines, storage or networking. By simplifying so much the change from IaaS to PaaS, and vice versa, for some modules, we are contributing to a real adaptation to our needs by minimizing the effort by the user. The underlying management tool will be in charge of the required provisioning and interoperation.

### Referencia Completa

**Título:** Bidimensional Cross-Cloud Application Management with TOSCA and Brooklyn

**Autores:** Jose Carrasco, Javier Cubo, Francisco Durán, Ernesto Pimentel

**Nombre de la conferencia:** 9th IEEE International Conference on Cloud Computing (CLOUD 2016), San Francisco, (EEUU) 2016. IEEE Computer Society.

**Año de la publicación:** 2016 (a aparecer).

### Indicios de Calidad

Este trabajo ha sido publicado en la conferencia internacional CLOUD, que trata sobre investigación en el área de Cloud Computing. Dicha conferencia se encuentra clasificada en el CORE B, según el ranking ERA de conferencias (2014): <http://portal.core.edu.au/conf-ranks/631/>. El artículo fue sometido a un proceso de revisión por parte de tres revisores externos.



# Reconfigurando Aplicaciones Multi-Cloud con Líneas de Producto Software Dinámicas\*

Javier Cubo, Nadia Gámez, Ernesto Pimentel, Lidia Fuentes

Dpto de Lenguajes y Ciencias de la Computación, Universidad de Málaga  
{cubo, nadia, ernesto, lff}@lcc.uma.es

**Resumen.** La reconfiguración dinámica de aplicaciones *multi-cloud* es un reto complejo aún no suficientemente explorado. En estos entornos las aplicaciones o sus módulos pueden estar desplegados en diferentes proveedores. Por lo tanto, reconfigurar en tiempo de ejecución estas aplicaciones puede requerir la modificación de la distribución en múltiples y heterogéneos proveedores. Obtener la nueva distribución para que sigan funcionando correctamente las aplicaciones no es una tarea sencilla, pues tanto los requisitos de las aplicaciones como las propiedades de los proveedores son muy diversos y variables. Además, la migración de las aplicaciones o sus módulos en tiempo real de un proveedor a otro puede conllevar problemas de compatibilidad y/o dependencias entre los módulos. Por lo tanto, el manejo de la variabilidad dinámica de las aplicaciones y proveedores, así como el de las dependencias existentes es deseable que se haga a un alto nivel de abstracción. Las Líneas de Producto Software Dinámicas (*DSLP*) utilizan modelos de variabilidad en tiempo de ejecución para obtener los cambios que han de llevarse a cabo durante la reconfiguración. En este trabajo, exploramos el uso del enfoque de *DSPL*, para que cuando ocurran problemas en los proveedores o se violen requisitos de las aplicaciones *multi-cloud*, las apps puedan ser reconfiguradas y seguir proporcionando los servicios adecuadamente a los usuarios.

**Palabras claves:** Reconfiguración; Aplicaciones Cloud; Multi-Cloud; DSPL; Variabilidad.

## 1 Motivación

En el paradigma de *Cloud Computing* existe una cierta variabilidad en los servicios, requisitos, propiedades y tecnologías soportadas por los diferentes proveedores *cloud*. Para solventar los problemas de portabilidad e interoperabilidad, existen soluciones para desplegar y gestionar las aplicaciones en entornos *multi-cloud*. Estas soluciones soportan el despliegue de aplicaciones o los módulos internos de una aplicación en diferentes proveedores. Una de estas soluciones es la plataforma *SeaClouds* (*Seamless adaptive multi-cloud management of service-based applications*, <http://seaclouds-project.eu>), desarrollada en el ámbito de un proyecto Europeo en el que dos de los autores de ese trabajo han participado activamente. Esta plataforma ayuda a los desarrolladores y administradores de aplicaciones *cloud*, haciendo más eficiente el diseño, desarrollo, planificación y gestión de las aplicaciones complejas en un entorno

\* Work supported by projects TIN2012-35669 and TIN2012-34840, funded by Spanish Ministry MINECO, FEDER; P11-TIC-7659, P09-TIC-5231 and P12-TIC1814 funded by Andalusian Gov; FP7-610531 and FP7-317731 funded by EU; and Univ. Málaga, Campus Exc. Int. Andalucía Tech.

heterogéneo y *multi-cloud*. De esta manera, cubre todos los aspectos de modelado, planificación y gestión para poder desplegar una aplicación o sus módulos en diversos proveedores que cumplan los requisitos de la aplicación, solventando las dificultades de portabilidad y de interoperabilidad existentes en los despliegues *multi-cloud*.

Sin embargo, estos entornos no son estáticos y en tiempo de ejecución pueden ocurrir problemas inesperados en los proveedores o se pueden violar ciertos requisitos, por lo que también es necesario soportar la reconfiguración dinámica de dichas aplicaciones. En el caso de estos entornos *multi-cloud*, esta reconfiguración se hace más compleja, ya que puede ser necesaria la migración de las aplicaciones (o de los módulos afectados) entre proveedores *cloud* compatibles. Para gestionar los cambios necesarios a llevar a cabo durante la reconfiguración dinámica, sería interesante modelar a un alto nivel la variabilidad existente tanto en los requisitos de las aplicaciones *multi-cloud* como en los proveedores *cloud* y las dependencias entre ellos. Las Líneas de Producto Software (*SPL*) y los Modelos de Variabilidad posibilitan la gestión de heterogeneidad en los sistemas software. De hecho, hay varios trabajos recientes que aplican el paradigma de las *SPLs* a los entornos *cloud* [1, 2, 3, 4]. Por ejemplo, en [1] proponen un análisis basado en *SPLs* para permitir la migración de aplicaciones tradicionales al *cloud*. Para ello modelan la variabilidad de los proveedores con Modelos de Características (*Feature Models*). De un modo similar [2] definen los Modelos de Características Cloud (*Cloud Feature Models*) para usarlos en el proceso de selección de servicios *clouds*, que filtra y reduce el número de configuraciones de servicios válidas para cada aplicación. Por otro lado, en [3] proponen modelar las aplicaciones basadas en *cloud* como familias de *Software-as-a-Service* y definen un marco de trabajo sistemático y unificado para modelar los servicios *cloud* de un modo independiente al proveedor. En el caso de [4], para entornos *multi-cloud* usan el modelado de la variabilidad ortogonal para manejar la configuración de aplicaciones *multi-cloud*. Todos estos trabajos muestran los beneficios de aplicar el enfoque de *SPLs* mediante el uso de los modelos de variabilidad en tiempo de diseño y despliegue de las aplicaciones *cloud*. Pero ninguno aborda el uso de estos modelos en tiempo de ejecución para ayudar a la reconfiguración de estas aplicaciones.

Las Líneas de Producto Software Dinámicas (*DSLPL*) producen software capaz de ser adaptado a cambios en tiempo real en el entorno. Las propuestas que usan este enfoque para reconfigurar los sistemas software, suelen utilizar los modelos de variabilidad en tiempo de ejecución para obtener los cambios que han de llevarse a cabo durante la reconfiguración. Así, nosotros, al igual que hacen los trabajos previamente mencionados con el enfoque de *SPL* en tiempo de diseño y despliegue, proponemos usar el enfoque de las *DSLPL* en tiempo de ejecución para manejar el proceso de reconfiguración de aplicaciones *multi-cloud*.

Entonces, en nuestra idea, además de usar los modelos de variabilidad para modelar los requisitos y las propiedades de las aplicaciones y las características de los proveedores, reflexionamos sobre la idoneidad de usarlos también en tiempo de ejecución. La idea es obtener automáticamente la nueva distribución de la aplicación en los proveedores para que siga funcionando correctamente en caso de producirse problemas o de violarse los requisitos. Las principales ventajas de usar los modelos de variabilidad para calcular la nueva distribución son dos. Por un lado, se pueden conservar fácilmente las especializaciones de esos modelos que contienen un



subconjunto de configuraciones posibles (es decir sólo las configuraciones válidas que respeten los requisitos de las aplicaciones). De este modo, cuando sea necesaria una reconfiguración se puede buscar la nueva configuración entre ese subconjunto sin tener que hacerlo desde cero. Y por otro lado, como en los modelos de variabilidad es posible definir restricciones y dependencias entre los diferentes elementos, usaremos esta característica para definir las dependencias entre los módulos que deban ser desplegados en el mismo proveedor o en proveedores compatibles. Esto nos permite, que si el problema o el error afecta solo a un módulo, se puede obtener el nuevo proveedor para ese módulo de forma local, sin tener que calcular la distribución para la aplicación entera. Pero al tener en cuenta las dependencias, es fácil detectar si, al hacer este cambio local a dicho módulo, es necesario hacer también cambios en otro módulo. A continuación presentamos la idea de nuestro enfoque y dado que estamos comenzando a discutir acerca de esta propuesta, varias cuestiones abiertas a ser discutidas serán presentadas.

## 2 Nuestro Enfoque y Plan de Trabajo

En la Figura 1, se muestra la visión general de nuestra propuesta. En esta se puede observar que se usan los modelos de variabilidad tanto para planificar la distribución de los módulos de una aplicación en los diversos proveedores como para obtener los cambios en esa distribución tras las sucesivas reconfiguraciones, aplicando el enfoque de *DSPLs*. Estos modelos de variabilidad pueden ser integrados en la plataforma *SeaClouds*. Esta plataforma ya proporciona herramientas de descubrimiento de servicios *cloud*, despliegue y monitorización en tiempo real para comprobar si cumplen los requisitos de la aplicación y que podrían lanzar una reconfiguración si fuera necesario. En este punto es cuando se usarían en tiempo de ejecución los modelos de variabilidad para obtener la nueva configuración, es decir, la nueva distribución de módulos de la aplicación en los diversos proveedores *cloud*.

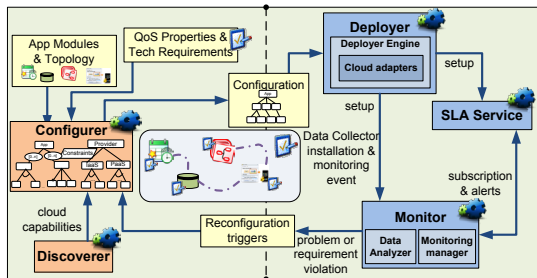


Figura 1. Proceso de Configuración y Reconfiguración de Aplicaciones Multi-cloud con Modelos de Variabilidad.

Para especificar los modelos de variabilidad pretendemos usar un lenguaje de modelado de la variabilidad propuesto como estándar, *Common Variability Language*

(CVL) [5]. Este lenguaje permite modelar la variabilidad (por ejemplo, mediante características opcionales o alternativas) de manera separada a un modelo base y tiene un conjunto de herramientas para manejar y analizar estos modelos. Dichas herramientas nos permitirán obtener automáticamente las configuraciones válidas que encajan con los requisitos de la aplicación. En nuestra propuesta, la reconfiguración se puede hacer de forma local al módulo con problemas sin tener que computar de nuevo todas las posibilidades, a diferencia de lo que actualmente se debe hacer con otras plataformas que dan la posibilidad de migrar módulos, como la funcionalidad proporcionada por *SeaClouds*.

Tras las ideas expuestas aquí, donde comenzamos a explorar el uso del enfoque de *DSPL* para abordar la reconfiguración dinámica de aplicaciones *multi-cloud*, queda aún mucho trabajo que llevar a cabo, tal como especificar todos los modelos de variabilidad, especialmente considerando una lista importante de los proveedores del mercado. Además, sería deseable automatizar por completo el proceso de configuración y reconfiguración, para lo que deberíamos poder trasladar la información de una configuración válida elegida que está modelada en CVL a un modelo de especificación estándar, como el que usa *SeaClouds*, TOSCA [7]. Una ventaja de tener este mapeado de CVL a TOSCA es que para la reconfiguración, detectando las diferencias [8] entre la configuración que estaba ejecutándose y la que se va a ejecutar, se puede obtener automáticamente qué partes del modelo TOSCA hay que cambiar. Por otro lado, en este trabajo sólo hemos mencionado que puede ser necesaria una reconfiguración dinámica cuando los proveedores no funcionan bien o no cumplen una serie de requisitos, pero hay más aspectos que pueden cambiar en tiempo de ejecución. Por ejemplo, sería deseable dar la posibilidad a los usuarios de cambiar los requisitos, como el coste de un servicio, o la disponibilidad o tiempo de respuesta de un proveedor *cloud*, y esto también lanzaría una reconfiguración para adaptar la aplicación a los nuevos requisitos.

## Referencias

1. García-Galán J., Rana O., Trinidad P. and Ruiz-Cortés A. (2013). Migrating to the Cloud - A Software Product Line based Analysis. In Proceedings of the 3rd International Conference on Cloud Computing and Services Science ISBN 978-989-8565-52-5, pages 416-426
2. Erik Wittern, Jörn Kuhlenkamp, and Michael Menzel. 2012. Cloud service selection based on variability modeling. In Proceedings of the 10th international conference on Service-Oriented Computing (ICSOC'12), Chengfei Liu, Heiko Ludwig, Farouk Toumani, and Qi Yu (Eds.). Springer-Verlag, Berlin, Heidelberg, 127-141
3. M. A. Matar, R. Mizouni and S. Alzahmi. "Towards Software Product Lines Based Cloud Architectures." Cloud Engineering (IC2E), 2014 IEEE International Conference on, Boston, MA, 2014, pp. 117-126.
4. Jamshidi, Pooyan and Pahl, Claus: Orthogonal Variability Modeling to Support Multi-cloud Application Configuration. In Advances in Service-Oriented and Cloud Computing: Workshops of ESOC 2014, Manchester, UK, September 2-4, 2014.
5. Haugen, Ø. et al.: Adding Standardized Variability to Domain Specific Languages. In Proc. of 12<sup>th</sup> Software Product Line Conference. IEEE Computer Society, (2008).
7. OASIS. TOSCA 1.0 (Topology and Orchestration Specification for Cloud Applications), Version 1.0, <http://docs.oasis-open.org/tosca/TOSCA/v1.0/TOSCA-v1.0.pdf>, 2013.
8. Gámez, N., Fuentes, L.: Architectural Evolution of FamiWare using Cardinality-Based Feature Models. Journal of Information and Software Technology 55(3): 563-580, (2013)

# Una aproximación DevOps para el Desarrollo Dirigido por Modelos de Servicios Cloud

Julio Sandobalín-Guamán<sup>1,2</sup>, Miguel Zúñiga-Prieto<sup>1,3</sup>, Emilio Insfran<sup>1</sup>, Sílvia Abrahão<sup>1</sup>, Carlos Cano<sup>1</sup>

<sup>1</sup>Departamento de Sistemas Informáticos y Computación, Universitat Politècnica de València, Camino de Vera, s/n, 46022, Valencia, España

{jsandobalin, mzuniga, einsfran, sabrahamo, carcage}@disc.upv.es

<sup>2</sup>Departamento de Informática y Ciencias de la Computación, Escuela Politécnica Nacional, Ladrón de Guevara, E11-253, P.O. Box 17-01-2759, Quito, Ecuador  
julio.sandobalin@epn.edu.ec

<sup>3</sup>Departamento de Ciencias de la Computación, Universidad de Cuenca, Av. 12 de Abril y Agustín Cueva, 01.01.168, Cuenca, Ecuador  
miguel.zunigap@ucuenca.edu.ec

**Resumen.** El desarrollo de software ágil promueve la entrega continua e incremental de software. Con cada incremento surge la necesidad de realizar una integración del nuevo código fuente con el existente para obtener una nueva versión del software. La nueva versión debe superar un conjunto de pruebas para ser desplegada en el entorno de operaciones. El tiempo que transcurre desde que el nuevo incremento pasa del entorno de desarrollo al entorno de operaciones debe ser minimizado para reducir costes económicos a las organizaciones. DevOps es un conjunto de principios y prácticas que optimizan el tiempo de entrega de un producto software, gestionan la infraestructura como código y mejoran la experiencia del usuario en base a la retroalimentación de sus comentarios. En trabajos anteriores hemos presentado DIARY como un método dirigido por modelos que soporta la reconfiguración dinámica de arquitecturas de servicios cloud producida por la integración incremental de nuevos servicios. En este trabajo presentamos una extensión del método DIARY con el fin de adoptar DevOps con una estrategia dirigida por modelos.

**Palabras Clave:** DevOps, Servicios Cloud, Integración y Reconfiguración de Servicios, Desarrollo Ágil, Desarrollo Dirigido por Modelos.

## 1 Introducción

El desarrollo ágil divide en iteraciones el desarrollo de un sistema software para obtener incrementos de software funcional. A pesar de los beneficios del desarrollo ágil, se originan nuevos retos debido a las brechas que aparecen al momento de integrar, realizar pruebas y desplegar a producción cada incremento. Además, el tiempo de entrega de cada incremento es un factor importante que origina una ventaja compe-

titiva a las organizaciones, permitiendo obtener retroalimentación de los clientes y de esta manera realizar una mejora continua de los productos de software.

Realizar manualmente los procesos de integración, pruebas y despliegue de incrementos de software ocasiona en la mayoría de los casos retrasos en la planificación de los proyectos y es propenso a errores con un resultado no confiable. DevOps [1] es un conjunto de principios y prácticas que optimizan el tiempo de entrega de software, gestionan la infraestructura como código y mejoran la experiencia del usuario en base a la retroalimentación de sus comentarios. Las prácticas de DevOps [2] son: i) la *integración continua* construye todo el software y ejecuta un conjunto de pruebas con cada incremento, ii) el *despliegue continuo* toma el software creado en la integración y lo despliega en un entorno de operaciones de una manera automática, y iii) la *entrega continua* proporciona la capacidad de liberar en el entorno de operaciones nuevas versiones de software varias veces al día.

DIARy [3] es un método de reconfiguración dinámica incremental de arquitecturas de servicios cloud basado en el Desarrollo de Software Dirigido por Modelos (DSDM). Este método tiene como objetivo apoyar a los desarrolladores de software en la integración incremental de servicios en entornos cloud, por tal motivo provee un conjunto de actividades que soportan: i) el diseño de la integración, ii) la implementación de la integración, y iii) el despliegue y reconfiguración arquitectónica.

La contribución de este trabajo es extender el método DIARy para utilizar las prácticas de DevOps y presentar un metamodelo para gestionar la infraestructura como código. El resto del artículo está estructurado de la siguiente manera: La Sección 2 presenta los trabajos relacionados. La Sección 3 presenta una extensión del método DIARy con soporte a las prácticas de DevOps. La sección 4 presenta un metamodelo para gestionar la infraestructura como código. Finalmente, en la Sección 5 se presentan las conclusiones y trabajos futuros.

## 2 Trabajos relacionados

TOSCA [4] especifica una topología y la orquestación para aplicaciones cloud, permite modelar nodos (máquinas físicas o virtuales) con sus respectivos componentes (base de datos, servidores web, etc.) para ser desplegados y orquestar las instancias como servicios en el cloud. En [5] se presenta una clasificación de los artefactos de la comunidad DevOps y como estos pueden ser transformados hacia TOSCA, los artefactos DevOps se clasifica en artefactos centrados en un nodo y centrados en un entorno. Los primeros son scripts, imágenes, etc. que pueden ser ejecutados en un único nodo como un servidor, una máquina virtual o un contenedor. Los últimos son scripts, plantillas, etc. que pueden ser ejecutados en múltiples nodos.

Wettinger et al. [6] [7] [8] [9] presenta una caracterización de los enfoques de despliegue de artefactos de software en un entorno de operaciones: i) se identifica el despliegue más eficiente de diferentes tipos de aplicaciones en soluciones IaaS (Infraestructure-as-a-Service) y PaaS (Platform-as-a-Service); ii) se presenta una interfaz que unifica la invocación de scripts de diferentes herramientas de despliegue de la comu-

nidad DevOps; y iii) se utiliza una aproximación al despliegue con snapshots de grano fino utilizando la virtualización de contenedores.

En [10] se presenta una aproximación para integrar el aprovisionamiento de la infraestructura con la gestión en el cloud dirigida por modelos. Se utiliza scripts de las herramientas Chef<sup>1</sup> y Jujú<sup>2</sup> como un Lenguaje de Dominio Específico para expresar el estado final del aprovisionamiento de hardware y el despliegue de software en el cloud.

Los esfuerzos de investigación se han centrado en el aprovisionamiento de infraestructura en el entorno de operaciones y en el despliegue de software, para tal fin se han utilizado las herramientas de la comunidad DevOps. Las propuestas de aprovisionamiento de infraestructura dirigida por modelos utilizan como Lenguaje de Dominio Específico los scripts de las herramientas de la comunidad DevOps, lo que origina el reto de definir un metamodelo que guíe el principio de infraestructura como código. El aporte de este artículo es presentar una extensión del método DIARY que utilice las prácticas de DevOps en los entornos de desarrollo y operaciones con un metamodelo para el aprovisionamiento de infraestructura.

### 3 Extensión de DIARY para dar soporte a DevOps

El método DIARY [3] [11] [12] tiene como objetivo sistematizar la reconfiguración dinámica de arquitecturas de servicios cloud producidas por la integración incremental de servicios y apoyar a los desarrolladores durante la fase de despliegue usando DSDM. Se debe agregar un conjunto de artefactos al método DIARY para que adopte las prácticas de DevOps. A continuación se detallan en cada práctica de DevOps la actividad y el conjunto de artefactos agregados para obtener la extensión de DIARY:

- **Integración Continua:** se agregó la actividad *Integración del Incremento* con los artefactos de entrada *Control de Versiones*, *Scripts de Construcción & Pruebas Unitarias*, *Modelo de Pruebas de Integración* y el artefacto de salida *Nuevo Incremento*.
- **Despliegue Continuo:** se agregó en la actividad *Despliegue & Reconfiguración Arquitectónica* como artefacto de entrada el *Modelo de Infraestructura* y como artefacto de salida los *Scripts de Aprovisionamiento de la Infraestructura & Despliegue*.
- **Entrega Continua:** con la adopción de las prácticas de *Integración Continua* y *Despliegue Continuo* la extensión del método DIARY tiene la capacidad de entregar varias versiones de incrementos de servicios cloud en periodos muy cortos de tiempo.

La Figura 1 muestra la extensión del método DIARY con soporte a las prácticas de DevOps. A continuación se detallan las actividades y artefactos de la extensión de DIARY:

---

<sup>1</sup> <https://www.chef.io>

<sup>2</sup> <https://jujucharms.com>

1. **Especificación de la Integración del Incremento:** Esta actividad tiene como objetivo permitir a los desarrolladores especificar como integrar un incremento de software en la arquitectura de un servicio cloud. Para ello permite a los desarrolladores especificar tanto la lógica de integración como el impacto arquitectónico de la integración, sin tomar en consideración las características específicas de un entorno cloud. En esta actividad los desarrolladores toman como entrada un incremento de software y lo integran como parte de un servicio siguiendo las *Guías para la Especificación de la Integración de Incrementos*. Las decisiones de diseño se toman en base a los términos de un *Acuerdo de Nivel de Servicios*. Esta es una actividad iterativa por lo que los desarrolladores podrían especificar la integración de incrementos de software compuesto por varios servicios. Con el fin de apoyar esta actividad los desarrolladores utilizan el Perfil de Especificación DIARy [12] que es un Lenguaje de Dominio Específico para especificar las decisiones de diseño de la integración, produciendo un *Modelo Extendido de la Arquitectura del Incremento*.
2. **Implementación del Incremento:** Esta actividad tiene como objetivo apoyar el proceso de integración mediante la generación de artefactos cloud y la implementación del incremento. Los artefactos a generar implementan: i) el *Protocolo de Interacción* entre los servicios cloud a ser integrados y la aplicación destino, ii) *Adaptadores Cloud* que corrigen incompatibilidades entre interfaces de servicios, y iii) *Artefactos Cloud* que implementan la lógica de servicios cloud.
3. **Integración del Incremento:** En esta actividad se realiza la Integración Continua de incrementos de una manera automática. Los incrementos que han sido implementados en la Actividad 2 son almacenados en un sistema de *Control de Versiones* para tener un control sobre las versiones de cada incremento y tener la capacidad de recuperar versiones anteriores. Para cada versión de cada incremento se debe ejecutar los *Scripts de Construcción & Pruebas Unitarias*, de manera que cada versión del incremento es un artefacto de software funcional que debe superar un conjunto de pruebas unitarias. En el caso, que un incremento tenga relación con uno o varios incrementos anteriores, se construyen dichos incrementos y se ejecutan un conjunto de pruebas de integración que son el resultado de una transformación de M2T a partir del *Modelo de Pruebas de Integración*. El resultado de esta actividad es un *Nuevo Incremento* que está listo para ser desplegado como un servicio en el cloud.
4. **Despliegue y Reconfiguración de la Arquitectura:** Los desarrolladores seleccionan del *Repositorio de Patrones de Adaptación* los patrones de adaptación apropiados para integrar la arquitectura del incremento en la arquitectura actual. Luego ejecutan las transformaciones de M2T con el fin de operacionalizar los patrones de adaptación seleccionados de acuerdo al *Modelo de Artefactos de Aplicaciones Cloud* y la tecnología de despliegue seleccionada. Los artefactos cloud generados son *Scripts de Reconfiguración de Servicios* que ayudan en la configuración y despliegue del incremento como un servicio en el cloud. Por otra parte, el equipo de operaciones realiza el Despliegue Continuo de incrementos con ayuda de: i) *Scripts de Reconfiguración de Servicios*, y ii) *Scripts de Aprovisionamiento de la Infraestructura & Despliegue*. El equipo de operaciones utiliza el *Modelo de Infraestructura* para modelar el hardware que será instalado y configurado en el entorno de

operaciones. El *Modelo de Infraestructura* se obtiene a partir del metamodelo para gestionar la Infraestructura como Código que se explica en la Sección 4. A partir del *Modelo de Infraestructura* se realizan transformaciones de M2T para obtener los *Scripts de Aprovisionamiento de Infraestructura & Despliegue* que son utilizados: i) para el aprovisionamiento de la infraestructura en el cloud, y ii) conjuntamente con los *Scripts de Reconfiguración de Servicios* realizan el despliegue del nuevo incremento en el entorno de operaciones como un servicio cloud.

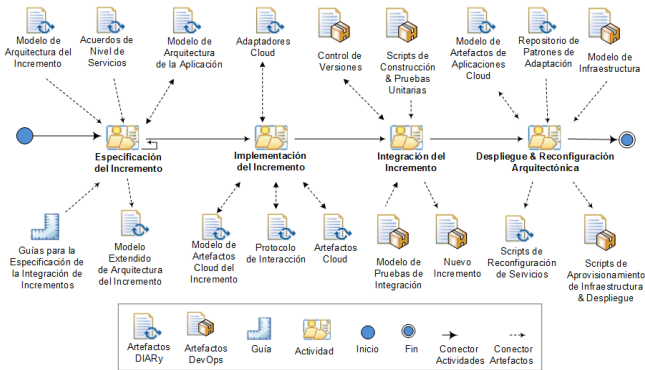


Figura 1. Extensión del método DIARy.

#### 4 Metamodelo para gestionar la Infraestructura como Código

El metamodelo propuesto en la Figura 2 proporciona un soporte al aprovisionamiento de infraestructura en el cloud mediante una estrategia dirigida por modelos. El metamodelo está concebido para ser extensible y parte de la creación de instancias de hardware, donde cada instancia (máquina virtual, volúmenes, etc.) tiene sus características (memoria RAM, procesador, disco duro, etc.) establecidas. Se parte del concepto que manejan los proveedores de servicios cloud, donde al crear una nueva instancia de una máquina virtual, las características del hardware vienen establecidas de acuerdo al nombre de cada instancia, por ejemplo en el caso de Amazon Web Services al seleccionar el nombre de la instancia *m4.large* sus características de hardware son 2 CPU y 8 GB de memoria. A continuación se explican las metaclases del metamodelo para gestionar la Infraestructura como Código:

- **File:** Representa los scripts que expresan el estado final del aprovisionamiento de hardware y el despliegue de software en el cloud. Los scripts tienen las sentencias para la instalación y configuración del hardware y software en el cloud.

- **Instance:** Es una metaclassa abstracta que representa las instancias de hardware (máquinas virtuales, volúmenes de datos, etc.). El atributo nombre (*name*) representa el nombre de las instancia de hardware que el proveedor cloud le asigna. Una instancia puede ser una máquina virtual de un proveedor cloud (*Machine*), una interconexión de varias instancias (*Network*) o un volumen de datos (*Volume*).
- **Command:** Es una metaclassa abstracta que representa un conjunto de sentencias (*Sentence*) que se deben ejecutar en las instancias de hardware creadas en los diferentes entornos cloud. El comando puede ejecutar una acción (*Run*), crear una variable de entorno (*Variable*) o agregar un archivo o carpeta (*Add*).

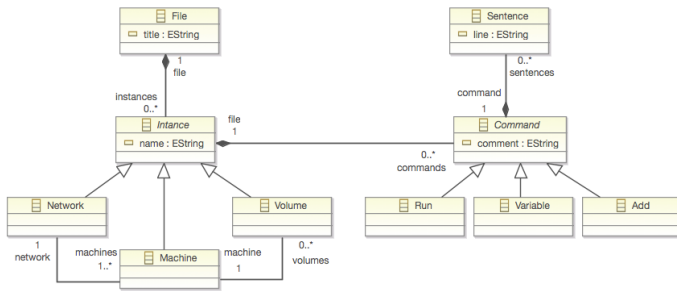


Figura 2. Metamodelo de Infraestructura.

La Figura 3 muestra un ejemplo de la implementación del aprovisionamiento de recursos de hardware para un servicio cloud. A continuación se detalla cada una de las instancias de hardware creadas:

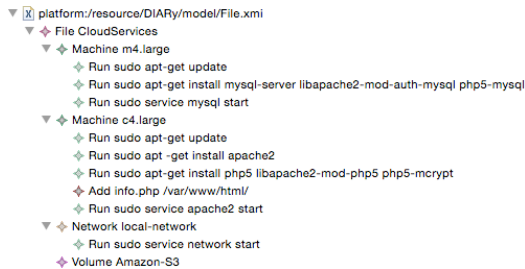


Figura 3. Modelo de Infraestructura.

- **Machine m4.large:** Es la imagen de una máquina virtual para un Servidor de Base de Datos. Tiene como características de hardware 2 CPU y 8 GB de memoria. Una



vez creada la instancia de hardware en la plataforma cloud, se ejecutan las siguientes sentencias:

```
# actualiza el sistema operativo
sudo apt-get update
# instala la base de datos MySQL
sudo apt-get install mysql-server libapache2-mod-auth-mysql php5-mysql
# inicia el servidor MySQL
sudo service mysql start
```

- **Machine c4.large:** Es la imagen de una máquina virtual para un Servidor Web. Tiene como características de hardware 2 CPU y 3,5 GB de memoria. Una vez creada la instancia de hardware en la plataforma cloud, se ejecutan las siguientes sentencias:

```
# actualiza el sistema operativo
sudo apt-get update
# instala el servidor Apache
sudo apt-get install apache2
# instala los módulos de PHP5
sudo apt-get install php5 libapache2-mod-php5 php5-mcrypt
# mueve el archivo info.php a la carpeta /var/www/html del servidor Apache
mv info.php /var/www/html/
# inicia el servidor Apache
sudo service apache2 start
```

- **Network local-network:** Es una red para la interconexión de las diferentes instancias creadas. Crea una conexión de red entre la *Machine m4.large* (Servidor Web) y la *Machine m4.large* (Servidor de Base de Datos).
- **Volumen Amazon-S3:** Es un volumen de almacenamiento para archivos que deben ser guardados por un largo periodo de tiempo. El volumen de datos está conectado con la *Machine m4.large* (Servidor Web).

#### 4.1 Soporte a la gestión de Infraestructura como Código

El caso de estudio se apoya en una aplicación para una compañía de manufactura presentado en [3][12]. A partir del concepto de Infraestructura como Código se desea aprovisionar de instancias de hardware un entorno cloud que permita el despliegue de un nuevo servicio (*incremento*) para distribuidores (*Dealers*) de forma que permita gestionar sus órdenes de pedido (*Place Order*) facilitando la interacción directa entre el sistema de tecnologías de la información de los distribuidores y el de la compañía de manufactura.

Las tareas de modelado de la arquitectura del servicio fueron realizadas utilizando Eclipse Modeling Framework [14]. Se desea instalar el servicio *Place Order* en un Servidor Web. El servicio *Place Order* está desarrollado en una arquitectura Java EE 1.6 y empaquetado en un archivo WAR. Se debe crear una instancia de hardware *m4.large* (2 CPU y 8 GB de memoria) en la plataforma cloud de *Amazon Web Services* con el sistema operativo Ubuntu LTS 14.04. Se debe instalar la *Java Virtual*

*Machine 1.6* y un el servidor web *Apache Tomcat 7*. Finalmente se debe desplegar el servicio *Place Order* en el servidor *Apache Tomcat 7*.

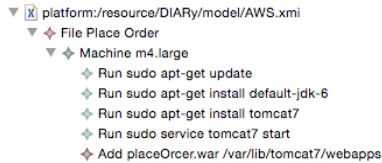


Figura 4. Modelo de Infraestructura para el servicio *Place Order*.

La Figura 4 muestra el *Modelo de Infraestructura* que soporta el aprovisionamiento de hardware para gestionar las órdenes de pedido (*Place Order*). A continuación se detalla cada uno de sus componentes:

- **Machine m4.large:** Es la imagen de una máquina virtual con un sistema operativo Ubuntu LTS 14.04 y con características de hardware: 2 CPU y 8 GB de memoria. Una vez creada la instancia de hardware en la plataforma cloud, se deben ejecutar las siguientes sentencias:

```
# actualiza el sistema operativo
sudo apt-get update
# instala la Java Virtual Machine 1.6
sudo apt-get install default-jdk-6
# instala el servidor web Apache Tomcat 7
sudo apt-get install tomcat7
# inicia el servidor de web Apache Tomcat 7
sudo service tomcat7 start
# coloca el archivo placeOrcer.war en la carpeta /var/lib/tomcat7/webapps
# con la finalidad de desplegar el servicio
mv placeOrcer.war /var/lib/tomcat7/webapps
```

Con transformaciones de M2T podemos generar un script para el aprovisionamiento de hardware en la plataforma Amazon Web Services y desplegar el servicio órdenes de pedido (*Place Order*) en el cloud. Este ejemplo, aunque preliminar, ayuda a determinar la viabilidad de esta propuesta y plantear el soporte al concepto de Infraestructura como Código para la extensión del método DIARy (Figura 1). Además, el *Modelo de Infraestructura* permite especificar tanto los recursos de hardware como el conjunto de comandos necesarios para obtener el *Script de Aprovisionamiento de la Infraestructura & Despliegue*. De esta forma, elevamos el nivel de abstracción y permitimos la generación automática de estos scripts por medio de transformaciones de M2T.

## 5 Conclusiones y trabajos futuros

En este artículo presentamos una primera aproximación de una propuesta dirigida por modelos para el aprovisionamiento de recursos en entornos cloud. Con la extensión del método DIARy se propone una solución holística a las prácticas DevOps de integración, despliegue y entrega continua de incrementos en los entornos de Desarrollo (Dev) y Operaciones (Ops). Se ha realizado una primera prueba de concepto para ilustrar algunas de las tareas de automatización para el aprovisionamiento de recursos de infraestructura de forma de estudiar la aplicabilidad y alcances de este enfoque en el entorno de operaciones.

Como trabajo futuro se pretende definir un *Lenguaje de Dominio Especifico* que capture los requisitos para la creación de instancias de hardware con la finalidad de generar sentencias específicas para los diferentes proveedores cloud de manera automática. De esta manera proponemos una solución a la diversidad de lenguajes de scripts existentes que dificultan una solución agnóstica hacia todos los proveedores de servicios cloud.

En resumen, la extensión del método DIARy presenta un solución basada en DSDM que permitirá automatizar de extremo-a-extremo la integración, implementación, pruebas y despliegue de un incremento de software (servicio) elevando el nivel de abstracción y permitiendo dar soporte al entorno de operaciones (Ops) por medio de la generación automática de los scripts de instalación y configuración de hardware.

Por último, tenemos previsto realizar casos de estudio más complejos y experimentos controlados que nos permitan validar la efectividad de la solución propuesta en cuanto a la integración y despliegue continuo de servicios en distintas plataformas cloud.

**Agradecimientos.** Investigación soportada por el proyecto Value@Cloud (MICINN TIN2013-46300-R), DIUC\_XIV\_2016\_038 y Microsoft Azure Research Awards.

## 6 Bibliografía

- [1] J. Wettinger, V. Andrikopoulos, and F. Leymann, “Enabling DevOps Collaboration and Continuous Delivery Using Diverse Application Environments,” *Move to Meaningful Internet Syst. OTM 2015 Conf.*, vol. 9415, pp. 348–358, 2015.
- [2] J. Humble and D. Farley, *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation*, 1 edition. Addison-Wesley Professional, 2010.
- [3] M. Zuñiga-Prieto, S. Abrahao, and E. Infran, “An Incremental and Model Driven Approach for the Dynamic Reconfiguration of Cloud Application Architectures,” *24th International Conference on Information Systems Development ISD2015*. 2015.

- [4] J. Wettinger, U. Breitenbücher, O. Kopp, and F. Leymann, "Streamlining DevOps automation for Cloud applications using TOSCA as standardized metamodel," *Futur. Gener. Comput. Syst.*, p. -, 2015.
- [5] J. Wettinger, U. Breitenbucher, and F. Leymann, "Standards-based DevOps automation and integration using TOSCA," in *Proceedings - 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing, UCC 2014*, 2014, pp. 59–68.
- [6] J. Wettinger, V. Andrikopoulos, S. Strauch, and F. Leymann, "Characterizing and evaluating different deployment approaches for cloud applications," in *Proceedings - 2014 IEEE International Conference on Cloud Engineering, IC2E 2014*, 2014, pp. 205–214.
- [7] J. Wettinger, T. Binz, U. Breitenbücher, O. Kopp, F. Leymann, and M. Zimmermann, "Unified Invocation of Scripts and Services for Provisioning, Deployment, and Management of Cloud Applications Based on TOSCA," in *Closer*, 2014, pp. 559–568.
- [8] J. Wettinger, U. Breitenbücher, and F. Leymann, "Compensation-Based vs. Convergent Deployment Automation for Services Operated in the Cloud," in *Service-Oriented Computing*, vol. 8831, X. Franch, A. Ghose, G. Lewis, and S. Bhiri, Eds. Springer Berlin Heidelberg, 2014, pp. 336–350.
- [9] J. Wettinger, K. Gorlach, and F. Leymann, "Deployment aggregates - A generic deployment automation approach for applications operated in the cloud," in *Proceedings - IEEE International Enterprise Distributed Object Computing Workshop, EDOCW*, 2014, pp. 173–180.
- [10] J. . Wettinger, M. . Behrendt, T. . Binz, U. . Breitenbücher, G. . Breiter, F. . Leymann, S. . Moser, I. . Schwertle, and T. . Spatzier, "Integrating configuration management with model-driven cloud management based on TOSCA," in *CLOSER 2013 - Proceedings of the 3rd International Conference on Cloud Computing and Services Science*, 2013, pp. 437–446.
- [11] M. Zuñiga-Prieto, J. Gonzalez-Huerta, S. Abrahao, and E. Insfran, "Towards a model-driven dynamic architecture reconfiguration process for cloud services integration," in *Models and Evolution Workshop CEUR Proceedings*, 2014, vol. 1331, pp. 52–61.
- [12] M. Zuñiga-Prieto, S. Abrahao, and E. Insfran, "Perfil UML para el Modelado de la Integración de Servicios Cloud en Procesos de Desarrollo Incremental," in *XI Jornadas de Ciencia e Ingeniería de los Servicios (JCIS)*, 2013.
- [13] N. Rathod and A. Surve, "Test orchestration a framework for Continuous Integration and Continuous deployment," in *2015 International Conference on Pervasive Computing: Advance Communication Technology and Application for Society, ICPC 2015*, 2015.
- [14] D. Steinberg, F. Budinsky, E. Merks, and M. Paternostro, *EMF: eclipse modeling framework*. 2008.

# Automatic Generation of Purchasing Plans for Cloud Services<sup>\*</sup>

Octavio Martín-Díaz, José María García, Pablo Fernandez, and Antonio Ruiz-Cortés

Dpto. Lenguajes y Sistemas Informáticos  
ETS. Ingeniería Informática – Universidad de Sevilla  
41012 Sevilla, España – Spain

omartindiaz@us.es, josemgarcia@us.es, pablofm@us.es, aruiz@us.es

**Abstract.** The myriad of cloud service providers, as well as their overwhelming variety of configuration and purchasing options, result in a highly complex purchasing scenario. Furthermore, users may specify their needs for cloud services provisioning with a certain scheduling restrictions. There is a need for an automatic support for obtaining an appropriate purchasing plan, which takes into account both service configurations and scheduling needs, while allowing the comparison among different providers and their various offerings. In this work, we present an automatic purchasing plan generator, which analyzes cloud service offerings from several providers to obtain an optimized purchasing plan according to user needs. From the obtained purchasing plan, our solution can provide the corresponding charge plan, possibly including discounts, which serves the purpose of comparing offerings to get the best option.

**Keywords:** Cloud Services, Purchasing Plan, Discounts

## 1 Introduction

Nowadays, IaaS purchasing aims to obtain virtual processing and storage resources from cloud providers, as a way to reduce cost if compared with the procurement of on-premise, private compute infrastructures. However, the myriad of cloud service providers, as well as their overwhelming variety of configuration and purchasing options [7], result in a highly complex purchasing scenario. In this setting, there are major heterogeneity issues that make the comparison among providers rather difficult, e.g. different variables for configurations, additional purchasing variants apart from the usual on-demand option, billing and charge processes, and particular discount rules, to name a few.

---

<sup>\*</sup> This work has been partially supported by the European Commission (FEDER), the Spanish and the Andalusian R&D&I programmes under grants TIN2015-70560-R, P12-TIC-1867, and P10-TIC-5906.

On the other hand, users may also specify their needs for cloud services provisioning including specific scheduling restrictions. These restrictions provide additional beforehand information concerning not only the number of instances of particular configurations that are needed at a certain time, but also the amount of time they are going to be used. Purchasing plans can be derived from the scheduling restrictions, which in turn may affect the charge plans proposed by cloud providers, since they usually offer discounts depending on purchase volume.

There are some tools which implement the search for an optimal configuration, such as [4] and [5], according to particular user needs. However, these tools do not take into account scheduling. As a consequence, their results, though very valuable in smaller cases, are but a simplified version of the overall purchasing scenario. Therefore, there is a need for an automatic support for obtaining an appropriate purchasing plan, which takes into account both service configurations and scheduling needs, while allowing the comparison among different providers and their various offerings.

In this work, we present an automatic purchasing plan generator, which analyzes cloud service offerings from several providers to obtain an optimized purchasing plan according to user needs. From the obtained purchasing plan, our solution can compute the corresponding charge plan. Our solution is able to search for optimal service configurations and apply discount rules from different providers, providing facilities to compare their offerings to select the best option. We have developed a prototype implementation that has been validated in a particular scenario with three different providers.

The rest of the article is structured as follows. Section 2 introduces a case study that further motivates our work. Next, Section 3 describes the conceptual model of the purchasing process. Then, Section 4 presents the architecture of our solution along with our validation results. Section 5 discusses the related work. Finally, Section 6 concludes the article and outline our future work.

## 2 Motivation

In order to further motivate our work, we present a case study about the *virtualization of laboratory classes* in the context of our Software Engineering courses. Laboratory classes may have a dynamic evolution from two viewpoints: (1) the software being used on those classes may evolve, usually requiring increasing computing resources, and thus possibly rendering the corresponding hardware obsolete at short notice; and (2) the demand, due to the number of students, may vary along the academic year. In order to increase flexibility and save overall costs, these classes can be virtualized by purchasing cloud infrastructure to support their dynamic environment.

In this context, we need to clearly state which particular computational requirements we have in order to obtain an appropriate purchasing plan that we have to follow with some cloud infrastructure services provider. As an example, let us consider that we need to provide infrastructure for the laboratory classes

of a four-month course beginning on Monday 3rd October 2016, which requires a very simple hardware configuration of single core CPU and 1GB memory (in short “Cpu:1 Mem:1”).

	MO	TU	WE	TH	FR
8h					
9h	G1 - 35		G3 - 25		G5 - 75
10h					
11h	G2 - 30		G4 - 30		G6 - 20
12h					

Fig. 1. Practices Scheduling

Figure 1 outlines the weekly sessions scheduling. There are 6 groups of students: G1 to G6. G1 to G4 have 4 hours-week (note that the sessions are overlapped between 10h and 11h), G5 has 2 hours-week, and G6 only has 1 hour-week. These groups have 35, 30, 25, 30, 75, and 20 students enrolled, respectively, which amount to 215 students. Additionally, there is a classroom open for students interested to further work outside regular sessions, which is planned to be open 6 hours-day from Monday to Friday for 10 students during 3 months, and 30 students during the 4<sup>th</sup> month which is closer to the exams period.

Starting with the stated user needs including scheduling restrictions, we can derive a simple purchasing plan that lists the necessary purchasing actions that have to be carried out to fulfill those needs. Table 1 shows each action as a temporal event, enumerating the number of instances to be purchased, the term (or number of months during which the purchasing is valid), and the expected usage (expressed as hours per month). In this plan we are considering overlaps between the regular sessions as shown in Figure 1.

In order to actually provision the infrastructural needs for these laboratory classes, we need to carry out those particular purchasing actions against a cloud infrastructure service provider. Before selecting a particular provider, different offers should be compared, considering actual charges and possible discounts to be applied. Consequently, there is a need for computing expected charges with respect to user needs to enable comparison amongst providers. Our solution analyzes user needs, their associated purchasing plans, and available service offerings to obtain a corresponding charge plan, hence allowing the user to choose the best option in each case.

**Table 1.** Purchasing plan for our case study

Event time	Instances (number)	Terms (months)	Usage (hours/month)
Mon Oct 03 2016	20	4	5
"	75	4	10
"	55	4	10
"	25	4	10
"	30	4	20
"	65	4	10
"	35	4	10
"	10	3	160
Mon Jan 03 2016	30	1	120

### 3 Modeling the Purchasing Process

As motivated in Section 2, we need a plan generator to compute a purchasing plan from the user needs which, in turn, given the provider’s offers, will determine the charge plan which, in the end, will make possible the comparison in order to select the most appropriate service. In order to automatically perform this process, we first need to model the relevant descriptions so that our solution can analyze and transform them into the resulting plan. In the following we present our modeling approach.

#### 3.1 User Needs

User needs specifies the client’s requirements on particular services (in our case study cloud infrastructure services, or IaaS in short). These requirements mainly state (1) the configurations which are needed to execute the client’s software, and (2) the expected usage schedule .

Figure 2 shows our conceptual model representing user needs. User needs have a reference to the client’s information, and a validity period which denotes the global time interval during which the IaaS is going to be used.

User needs are composed of several scheduling groups, which represent the particular usage schedule of certain service configuration [7].Essentially, each scheduling group is composed of scheduling items, which are “*temporal composites*” that detail the validity periods and number of instances of the same configuration that are needed. The validity period of a scheduling item is a time interval that may be periodic, and possibly disjoint or overlapped with others.

#### 3.2 Purchasing and Charge Plans

A purchasing plan is composed of purchasing events. Each event comprises a series of purchasing actions that have to be performed at the same point of time. Conversely, each purchasing action is represented in Figure 3, including



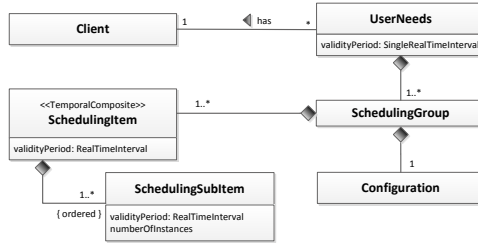


Fig. 2. Conceptual model for user needs

the required service configuration (i.e. memory and CPU in our case study), the purchasing type (prepayment, reservation, on-demand, or others), the expected usage (in hours/month), the number of instances to be purchased, and the term (number of months during which the purchase is going to be effective).

Initially, a purchasing plan is not associated with a particular provider. However, when searching for the most appropriate configuration from provider offerings, there could be specific information (e.g. base price) that would be stated by corresponding providers. Consequently, the generation of the specific purchasing plan and subsequent charge plan needs to be particularized for each provider.

Similarly, a charge plan is composed of charge events. In this case, our model specifies the point of time at which the actual charge is carried out, the type of charge (e.g. monthly payment or one-time payment), the monetary quantity, and a reference to the purchasing information that the charge is related to. Though it is not represented in the figure, purchasing and billing policies, including discount rules, are taken into account when generating charge plans from the purchasing plan, as discussed in the following section.

## 4 Prototype Implementation

In order to validate our approach, we developed a prototype solution that is based on the models described in Section 3. This solution is able to automatically derive purchasing plans and corresponding charge plans from different providers with regards to stated user needs. Figure 4 sketches the overall architecture of our solution.

On the one hand, service offerings are automatically imported in our system using a JSON-LD [10] parser that takes JSON files published by service

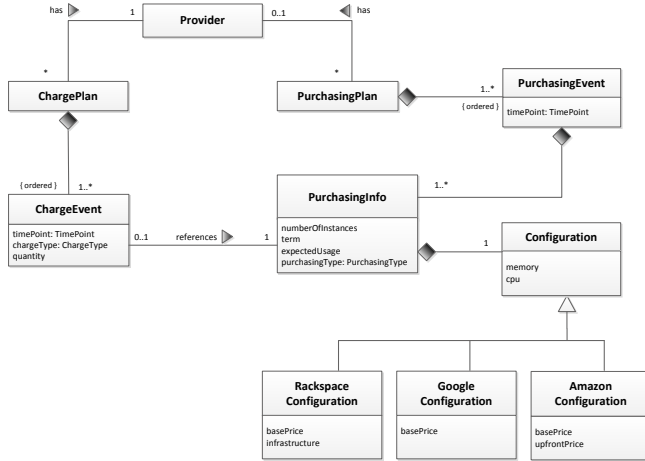


Fig. 3. Conceptual model for purchasing and charge plans

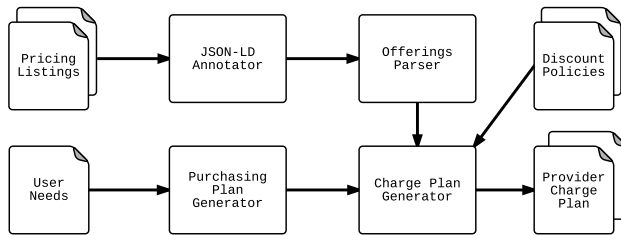


Fig. 4. Architecture of our automatic purchasing plan generator

providers, such as Amazon<sup>1</sup> or Google<sup>2</sup>, and annotate some properties to identify common properties (such as base price, CPU, or memory) using JSON-LD facilities. We use the conceptual model sketched in Figure 3 as the fundamental schema to annotate pricing information from these providers, enabling interoperability of their original JSON schemas. Then, annotated JSON pricing listings are parsed in order to populate the catalog of service offerings from different providers.

On the other hand, user needs are instantiated according to the model discussed in Section 3. The purchasing plan generator component analyzes user needs and scheduling restrictions in order to obtain a specific purchasing plan, such as the one shown in Table 1. Generic purchasing plans need to be instantiated with respect to a specific service provider, so that a subsequent charge plan is generated. The charge plan generator component of our system perform this process, taking into account discount policies of each provider to finally obtain a specific provider charge plan that completely fulfill user needs.

In order to validate our solution, we carried out the case study described in Section 2 using the implemented prototype. We considered service offerings from Amazon, Google, and Rackspace. As stated by our user needs, our tool searched for the closer configuration to “Cpu:1 Mem:1” in their catalogs, using the approach presented in [7]. According to the *on demand* purchasing type, Table 2 summarizes the results of the search<sup>3</sup>:

**Table 2.** Selected configurations for our case study

Provider	Configuration	Base price	Infrastructure Price
Amazon	t2.micro	0.013\$/hour	
Google	g1-small	0.027\$/hour	
Rackspace	General1-1	0.032\$/hour	0.005\$/hour (Min 50\$/month)

After applying each provider’s policies regarding the purchasing process, including discount rules, Table 3 shows the corresponding charge plans:

In this case, Amazon is the best option. However if we elevate the requirements, for instance increasing the usage to 15 or 24 hours-day, requiring a more powerful configuration “Cpu:16 Mem:120” or “Cpu:32 Mem:240”, or increasing the number of students, then it would be possible that the application of discount rules results in other providers becoming better options. Table 4 shows the resulting totals.

In this situation the best option depends on the particular case. Amazon is cheaper with smaller configuration, but it is the most expensive when strong requirements are needed, with Google offerings being the best positioned.

<sup>1</sup> <http://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/price-changes.html>

<sup>2</sup> <https://cloudpricingcalculator.appspot.com/static/data/pricelist.json>

<sup>3</sup> Note that all prices considered were obtained on April, 20<sup>th</sup> 2016.

**Table 3.** Charge plans for our case study

Amazon		Google		Rackspace	
Date	Quantity	Date	Quantity	Date	Quantity
Nov 01 2016	56.74\$	Nov 02 2016	117.85\$	Nov 03 2016	202.70\$
Dec 01 2016	63.05\$	Dec 02 2016	130.95\$	Dec 03 2016	202.70\$
Jan 01 2017	63.05\$	Jan 02 2017	130.95\$	Jan 03 2017	202.70\$
Feb 01 2017	86.45\$	Feb 02 2017	179.55\$	Feb 03 2017	276.70\$
Mar 01 2017	8.91\$	Mar 02 2017	18.50\$		
<b>Total</b>	<b>278.20\$</b>	<b>Total</b>	<b>577.80\$</b>	<b>Total</b>	<b>884.80\$</b>

**Table 4.** Comparison between different variants of our case study

Case study variant	Amazon	Google	Rackspace (Prepayment)
4 hours-day (initial case study)	278\$	577\$	884\$
9 hours-day	947\$	1,958\$	2,697\$
15 hours-day ...	1,579\$	3,218\$	4,495\$
24 hours-day ...	2,527\$	4,889\$	7,192\$
... and Cpu:16 Mem:120	536,544\$	185,528\$	238,157\$
... and Cpu:32 Mem:240	1,073,088\$	365,057\$	449,254\$
... and Cpu:32 Mem:240 / x10 users	10,730,880\$	3,650,573\$	4,262,025\$

As a conclusion, obtaining the best provider for certain user needs and scheduling restrictions depends on the actual configurations needed as well as the discount policies that providers offer. Our solution supports this process, facilitating the computation of purchasing and charge plans that can be compared among different providers.

## 5 Related Work

As far as we know, there is no approach to the purchasing problem which takes into account both heterogeneity of configuration, purchasing options, and discount rules, together with a required usage scheduling. However, there are related works which partially solve the purchasing and related problems.

First, previously cited tools such as [4, 5] compare providers given a configuration and some purchasing options, but discount rules and scheduling are not taken into account. From a workflow perspective, there are approaches to optimize the IaaS purchasing process supporting a workflow under scheduling restrictions [1, 6, 12, 13]. From an economic perspective, some studies on the cost of cloud services have been carried out [2, 3, 15], focussed on the analysis of investments to support the decision making during the purchasing process, but they do not address how the different purchasing options are formally expressed. From a quality perspective, in [8] authors provide a comprehensive model to

automate the ranking of different providers, but define a simplified version of cost and do not introduce a real purchasing option variability in their scenario.

Concerning purchasing models, similar limitations can be found in other related works such as [9] where authors present an initial review of different pricing models for cloud services that shows simple cost model scenarios, but elements such as purchasing processes variability or discounts are not discussed. For instance, in [16], authors present a formal language for cloud services.

There are also general comparisons among cloud providers, such as [14] whose author presents a comparison framework where different characteristics of the purchasing process, including the price model, are evaluated in order to obtain a classification of service providers.

## 6 Conclusions and Future Work

Purchasing plans are of utmost importance when trying to optimize computational resources required to fulfill some user needs during specific time periods. This article presents a solution to automatically derive purchasing plans from user needs specification including scheduling restrictions. After modeling user needs for a particular scenario, our prototype implementation searches for appropriate service configurations from different providers and generates the corresponding charge plan related to the purchasing actions needed to fulfill user requirements. Consequently, users can choose the best provider with respect to their needs.

As future work, we plan to further validate our approach, automatically crawling pricing and service configuration options from other providers, such as Microsoft or Heroku. Moreover, we will integrate our current solution with previous work in order to optimize purchasing plans and hence obtain more competitive charge plans [11]. Finally, we are evaluating rule-based approaches to generalize the definition of discount rules from service providers.

## Acknowledgements

Authors would like to thank Manuel Arenillas for his support on the prototype implementation.

## References

- [1] Anbazhagi, Latha Tamilselvan, and Shakkeera. QoS based Dynamic Task Scheduling in IaaS Cloud. In *2014 IEEE International Conference on Recent Trends in Information Technology (ICRTIT)*, 2014.
- [2] Slaven Brumec and Neven Vrček. Cost effectiveness of commercial computing clouds. *Information Systems*, 38(4):495–508, 2013.
- [3] Se-Hak Chun and Byong-Sam Choi. Service models and pricing schemes for cloud computing. *Cluster Computing*, 17(2):529–535, 2014.

- [4] Clouddorado.com. Cloud Computing Price Comparison Engine. <http://www.clouddorado.com/>.
- [5] CloudScreener.com. Cloud Computing Comparison and Evaluation. <http://www.cloudscreener.com/>.
- [6] Ruben Van den Bossche, Kurt Vanmechelen, and Jan Broeckhove. Cost-Optimal Scheduling in Hybrid IaaS Clouds for Deadline Constrained Workloads. In *2010 IEEE 3<sup>rd</sup> International Conference on Cloud Computing*, 2010.
- [7] Jesús García-Galán, Pablo Trinidad, Omer F. Rana, and Antonio Ruiz-Cortés. Automated Configuration Support for Infrastructure Migration to the Cloud. *Future Generation Computer Systems*, 2015.
- [8] Saurabh Kumar Garg, Steve Versteeg, and Rajkumar Buyya. A framework for ranking of cloud computing services. *Future Generation Computer Systems*, 29(4):1012–1023, 2013.
- [9] Sahil Kansal, Gurjeet Singh, Harish Kumar, and Sakshi Kaushal. Pricing models in cloud computing. In *ACM International Conference on Information and Communication Technology for Competitive Strategies (ICTCS)*, page 33, 2014.
- [10] Markus Lanthaler and Christian Gütl. On using JSON-LD to create evolvable restful services. In Rosa Alarcón, Cesare Pautasso, and Erik Wilde, editors, *Third International Workshop on RESTful Design, WS-REST '12, Lyon, France, April 16, 2012*, pages 25–32. ACM, 2012.
- [11] O. Martín-Díaz, P. Fernández, P. Trinidad, and A. Ruiz-Cortés. Apoyo a la toma de decisiones en la compra de iaas. In *10<sup>th</sup> Jornadas de Ciencia e Ingeniería de Servicios (JCIS'14)*, pages 179–188, Cádiz, Spain, 2014.
- [12] Nuttapon Netjinda, Booncharoen Sirinaovakul, and Tiranee Achalakul. Cost Optimal Scheduling in IaaS for Dependent Workload with Particle Swarm Optimization. *The Journal of Supercomputing*, 68(3):1579–1603, 2014.
- [13] M. K. Nivodhini, K. Kousalya, and S. Malliga. Algorithms to Improve Scheduling Techniques in IaaS Clouds. In *2013 IEEE International Conference on Information Communication and Embedded Systems (ICICES)*, 2013.
- [14] Thoran Rodrigues. 11 Cloud IaaS Providers Compared. <http://www.techrepublic.com/blog/the-enterprise-cloud/11-cloud-iaas-providers-compared/>.
- [15] Bhanu Sharma, Ruppa K Thulasiram, Parimala Thulasiraman, Saurabh K Garg, and Rajkumar Buyya. Pricing cloud compute commodities: a novel financial economic model. In *12<sup>th</sup> IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*, pages 451–457, 2012.
- [16] Rafael B. Uriarte, Francesco Tiezzi, and Rocco De Nicola. SLAC: A formal Service-Level-Agreement Language for cloud computing. In *IEEE/ACM<sup>th</sup> Int. Conf. on Utility and Cloud Computing (UCC)*, pages 419–426, 2014.

# Governing the service-chain: Challenges ahead

Pablo Fernandez

University of Sevilla  
pablofm@us.es

**Abstract.** As Information Systems are evolving into an ecosystem of services, organizations face the persistent challenge of IT governance. In such a context, Cloud Computing shift has supported a growing service-chain that has transformed the business model from industry. In this position paper we outline the dimensions of this service chain reality and the role of Service Level Agreement as a foundation to support its governance challenges.

**Keywords:** service-chain,governance,sla

## 1 Introduction

In spite Service-oriented approach has been a success, its seminal orchestration model has been clearly questioned: Originally, SOC proposed a centralized governance approach mainly based on service buses and heavy middle-wares but industry trends clearly show a shift towards lightweight fine-grained services that are easier to replicate, evolve and test (i.e the Micro-services architecture <sup>1</sup>). In such a context, information systems in organizations face a new reality where traditional boundaries of applications are blurring to rather conform ecosystems of services that should be governed in a coherent and consistent way.

Moreover, the advent of Cloud Computing approach has boosted organizational integration where Infrastructures and Platforms are offered as services conforming a new commodity that is increasingly used in all sort of scenarios. This new model of externalized infrastructure makes it easy to overcome the traditional limitation of physical infrastructures to provide elasticity mechanisms and adapt to a changing environment.

As a consequence, from a business model perspective this reality of a highly integrated ecosystem of services, has paved the way for an important transformation: Information systems are no longer meant to be isolated entities with specific integration points that support an specific business processes; alternatively, Information Systems are starting to conform a global service chain where all components are both consumers and providers of services and where virtualized infrastructures and end-users are the terminals of the chain.

This service-chain can be characterized in different dimensions in order to analyze its current status and evolution:

---

<sup>1</sup> <http://martinfowler.com/articles/microservices.html>

- Taking the tiers defined in Cloud Computing paradigm (IaaS, PaaS and SaaS) as an starting point, we can see two growth directions in service-chains:
  - **Vertical** Where service providers of inferior tiers (IaaS/PaaS) support to superior tiers (PaaS/SaaS).
  - **Horizontal** Where a functional integration is developed within the same tier in order to extend the service provided to a superior tier.
- From an organizational perspective, service-chains are applied in two scenarios (that could be integrated) with different governance goals:

In such a context, these scenarios and growth dimensions demand a higher level of coordination that could support the evolution and governance in a more automated way. These key challenges can be addressed by using the concept of an operational Service Level Agreement (SLA)[1] as a corner-stone to articulate the ecosystem of services that conform the service-chain [3]. Specifically, operational SLA would require a rich model and a supporting infrastructure that could include aspects such as Costs, Service Descriptions, Configuration properties [2], Guarantees or Compensations [6][5]; i.e. an evolved idea of SLAs as a digital artifact that can be instrumented and managed in an automated way.

As a promising consequence of this evolved SLA, we would be in the verge to integrate business models as an operational asset that would allow a leverage of information systems from a pure technical level to a business-driven ecosystem of services [4]. Moreover, this shift would be a foundation for new generations of service-chains that face key governance challenges to transform ecosystems in different dimensions:

- **Elasticity**, to adapt and recover from extra-ordinary situations.
- **Robustness**, to estimate and control potential risks.
- **Efficiency**, to optimize the operational processes and internal resource usage to deliver the service.

Summing up, these challenges represent an interesting horizon where service community should embrace the reality of an evolving ecosystem of services that needs better governance models and incorporate elements (such as the SLA) to develop novel business-driven systems integrated in a global service-chain.

## References

1. A. Andrieux, K. Czakowski, A. Dan, K. Keahey, H. Ludwig, T. Nakata, J. Pruyne, J. Rofrano, S. Tuecke, and M. Xu. Web Services Agreement Specification (WS-Agreement) Version 1.1 draft 20, September 2006.
2. P. Fernandez, Hong-Linh Truong, Schahram Dustdar, and Antonio Ruiz-Cortés. Programming elasticity and commitment in dynamic processes. *Internet Computing, IEEE*, 19(2):68–74, Mar 2015.
3. G. Grabarnik, H. Ludwig, and L. Shwartz. Management of service process qos in a service provider - service supplier environment. In *E-Commerce Technology and the 4th IEEE International Conference on Enterprise Computing, E-Commerce, and E-Services, 2007. CEC/EEE 2007. The 9th IEEE International Conference on*, pages 543–550, July 2007.



4. P. Leitner, W. Hummer, B. Satzger, C. Inzinger, and S. Dustdar. Cost-efficient and application sla-aware client side request scheduling in an infrastructure-as-a-service cloud. In *Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on*, pages 213–220, June 2012.
5. Carlos Müller, Antonio M Gutiérrez, Octavio Martin-Diaz, Manuel Resinas, Pablo Fernandez, and Antonio Ruiz-Cortés. Towards a formal specification of slas with compensations. In *On the Move to Meaningful Internet Systems: OTM 2014 Conferences*, pages 295–312. Springer Berlin Heidelberg, 2014.
6. Carlos Müller, Manuel Resinas, and Antonio Ruiz-Cortés. Automated analysis of conflicts in ws-agreement documents. *IEEE Transactions on Services Computing*, 7(4):530–544, October-December 2014.



# Elasticidad y variabilidad en procesos de negocio



# VIVACE: A framework for the systematic evaluation of variability support in process-aware information systems\*

Clara Ayora<sup>1</sup>, Victoria Torres<sup>1</sup>, Barbara Weber<sup>2</sup>, Manfred Reichert<sup>3</sup>, and Vicente Pelechano<sup>1</sup>

<sup>1</sup> Universitat Politècnica de València, Spain  
Pros Research Center

`cayora,vtorres,pele@pros.upv.es`

<sup>2</sup> University of Innsbruck, Austria

`barbara.weber@uibk.ac.at`

<sup>3</sup> University of Ulm, Germany  
`manfred.reichert@uni-ulm.de`

## 1 Summary of the Contribution

Each product an enterprise develops or produces and each service it delivers to its customers result from the coordinated execution of a set of activities (i.e., business functions). In this context, business processes act as the drivers enabling this coordination [1]. Consequently, process-aware information systems (PAISs) provide a guiding framework in enterprise computing, supporting the entire business process lifecycle [2], managing, executing, and analyzing the internal business processes of an enterprise (e.g., sales business processes) based on explicitly specified process models. The increasing adoption of PAISs in enterprises during the last decade has resulted in large process model repositories [3-5]. Usually, such repositories comprise collections of related process model variants (aka process family) which may pursue the same or similar business objective (e.g., product sales, patient treatment, or car maintenance) but at the same time differ in several aspects due to their varying application context (e.g., regulations to be obeyed in different countries or the type of product to be delivered). As a result process families may comprise dozens or hundreds of process variants [6].

In order to efficiently and effectively manage process families, enterprises are interested in capturing common process knowledge only once and in making it reusable in terms of a reference process model [7]. However, the lack of support for explicitly describing variations in process families in off-the-shelf process modeling suites has motivated the development of approaches that treat variability as a first class citizen. These approaches contribute to avoid model redundancies, foster model reusability, and reduce modeling efforts. However, explicitly capturing and modeling variability introduces additional complexity with

---

\* This work has been developed with the support of MINECO under the SMART-ADAPT TIN2013-42981-P project and co-funded by the ERDF

respect to the design of the process modeling language used. In order to make these process variability approaches amenable for industrial use, the quality of process variant models becomes crucial. In particular, this necessitates proper assistance of process engineers with respect to the modeling and management of process variability. This means that process engineers should be supported in selecting that process variability approach fitting best to their needs. However, there is a lack of profound methods for systematically assessing and comparing existing process variability approaches. Although several attempts to characterize process variability have been made, a comprehensive and well elaborated framework for assessing process modeling tools and PAISs regarding their ability to deal with process variability is still missing.

Therefore, in this paper we present results from a systematic literature review (SLR) of approaches enabling process variability along the process lifecycle. Besides elaborating the state of the art, we want to systematically analyze and assess existing process variability approaches regarding their expressiveness with respect to process variability modeling as well as their support for managing variability along the process lifecycle. In addition, based on the empirical evidence provided by the SLR, we derive the VIVACE framework, which shall allow for the systematic assessment and comparison of existing process variability approaches. This framework enables process engineers to select that variability approach meeting their requirements best as well as helps them in implementing PAISs supporting process variability.

## References

1. Weske, M.: *Business Process Management - Concepts, Languages, Architectures*, 2nd Edition. Springer (2012)
2. Weber, B., Sadiq, S.W., Reichert, M.: Beyond rigidity - dynamic process lifecycle support. *Computer Science - R&D* **23** (2009) 47–65
3. Rosemann, M.: Potential pitfalls of process modeling: part A. *Business Proc. Manag. Journal* **12** (2006) 249–254
4. Dijkman, R.M., Rosa, M.L., Reijers, H.A.: Managing large collections of business process models - current techniques and challenges. *Computers in Industry* **63** (2012) 91–97
5. Galster, M., Lapre, L., Avgeriou, P.: SOA in variability-intensive environments: Pitfalls and best practices. *IEEE Software* **31** (2014) 77–84
6. Müller, D., Herbst, J., Hammori, M., Reichert, M.: IT support for release management processes in the automotive industry. In: *Business Process Management, 4th International Conference, BPM 2006, Vienna, Austria, September 5-7, 2006, Proceedings*. (2006) 368–377
7. Reichert, M., Weber, B.: *Enabling Flexibility in Process-Aware Information Systems - Challenges, Methods, Technologies*. Springer (2012)

# Reparación de alineamientos en modelos de proceso similares

B. Vázquez-Barreiros<sup>1</sup>, S.J. van Zelst<sup>2</sup>, J.C.A.M. Buijs<sup>2</sup>, M. Lama<sup>1</sup>, M. Mucientes<sup>1</sup>

<sup>1</sup> Centro Singular de Investigación en Tecnoloxías da Información (CiTIUS)  
Universidade de Santiago de Compostela, Spain

{borja.vazquez, manuel.lama, manuel.mucientes}@usc.es

<sup>2</sup> Department of Mathematics and Computer Science  
Eindhoven University of Technology, The Netherlands

{s.j.v.zelst, j.c.a.m.buijs}@tue.nl

**Resumen** Una de las ramas más importantes de la minería de procesos es el *análisis de conformidad*, es decir, analizar hasta qué punto un proceso de negocio se ajusta a los datos de ejecución de los procesos de negocio observados. Los *alineamientos* son el instrumento estándar *de facto* dentro de las analíticas de conformidad. Los alineamientos relacionan elementos de un registro de eventos con las actividades presentes en un modelo de proceso. Sin embargo, calcularlos es un problema combinatorio y, por lo tanto, extremadamente costoso. En este artículo mostramos cómo es posible reparar un alineamiento para un proceso dado, utilizando como punto de partida un modelo y un alineamiento previo. De este modo, demostramos como es posible obtener un nuevo alineamiento, en un tiempo significativamente mejor, reparando aquellas partes que ya no se ajustan al comportamiento del nuevo modelo.

## 1. Introducción

La minería de procesos tiene como objetivo el análisis, entendimiento y mejora de procesos de negocio. Una de las ramas más importantes de la minería de procesos es el *análisis de conformidad*, es decir, analizar hasta qué punto un proceso de negocio se ajusta a los datos de ejecución de los procesos de negocio reales. Los *alineamientos* [1] han demostrado ser muy eficaces para el análisis de conformidad. En esencia, un alineamiento alinea un registro de eventos con un modelo de proceso, permitiendo aplicar una gran variedad de técnicas de análisis que dan lugar a diferentes estadísticas que describen esta relación entre el registro de eventos y el modelo de proceso.

Dado que son un problema combinatorio, el uso de alineamientos en conjunto con registros de eventos de tamaño considerable generalmente se traduce en un empeoramiento en los tiempos de ejecución. Por lo tanto, la enorme complejidad de calcular alineamientos tiene sus efectos sobre las técnicas que los utilizan internamente [1,2,3,4]. Sin embargo, algunas de estas técnicas comparten una interesante propiedad: el análisis de modelos de proceso similares. Por ejemplo,

en los algoritmos de descubrimiento basados en técnicas evolutivas, cada generación de modelos de proceso se crea a partir de ligeras manipulaciones de la actual generación de modelos de procesos [5]. Por otro lado, el Inductive Visual Miner [4] permite al usuario aplicar técnicas de filtrado, lo que resulta en un nuevo, ligeramente similar, modelo de proceso sobre el cuál es necesario volver a calcular los alineamientos. Por lo tanto, surge la pregunta de si es posible utilizar alineamientos previamente calculados como base para el cálculo de los nuevos alineamientos.

En este artículo proponemos un método para *reparar alineamientos* donde, dado un nuevo modelo de proceso y un alineamiento calculado en un modelo de proceso diferente, calcula un nuevo alineamiento para el nuevo modelo de proceso. Esta técnica identifica los fragmentos del alineamiento ya existente que no se corresponden con el nuevo modelo de proceso, y los reemplaza con nuevos fragmentos de alineamientos que sí corresponden. Debido a que este método sólo se centra en aquellas partes del alineamiento que no encajan, el tiempo de cálculo disminuye de manera significativa. Además, mostramos que la pérdida de optimalidad es limitada y se mantiene dentro de límites aceptables.

El concepto presentado en este trabajo, es decir, la resolución de un subproblema en lugar de todo el problema, es similar a los métodos que apuntan a la descomposición de las técnicas de minería de procesos [6,7,8]. La principal diferencia es que la técnica presentada en este artículo da lugar a un alineamiento para toda la traza y para todo el modelo de proceso al completo, mientras que las técnicas de descomposición previamente citadas suelen proporcionar soluciones para subproblemas, y no para la solución completa.

El resto del artículo se organiza en las siguientes secciones. La Sección 2 introduce los conceptos de alineamientos y árboles de proceso. Luego, la Sección 3 describe el proceso de reparación. En la Sección 4 se presentan los resultados y la Sección 5 finaliza con las conclusiones del trabajo.

## 2. Preliminares

En esta sección se introduce la noción de *árboles de proceso* y *alineamientos*.

### 2.1. Árboles de proceso

Un *árbol de proceso* [3,9] es una representación jerárquica de un flujo de trabajo estructurado. Las hojas de un árbol de procesos están etiquetados como *actividades*, mientras que los nodos internos se etiquetan como operadores, que se utilizan para especificar la relación entre sus hijos. Existen cinco operadores estándar [3]: el operador secuencial ( $\rightarrow$ ), el operador de ejecución paralela ( $\wedge$ ), el operador de selección exclusiva ( $\times$ ), el operador de selección inclusiva ( $\vee$ ) y el operador de lazo ( $\odot$ ). Los operadores pueden tener un número arbitrario de hijos en cualquier orden, excepto el operador secuencial y de lazo. Para el operador secuencial, el número de hijos puede ser aleatorio, pero el orden de los hijos especifica el orden en el que se deben evaluar, es decir, de izquierda a derecha.



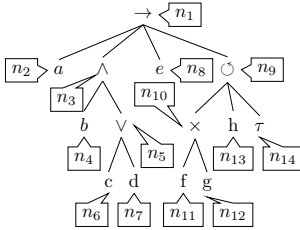


Figura 1. Árbol de proceso  $P_1$ .

$\langle n_2, n_7, n_4, n_8, n_{11}, n_{14} \rangle \equiv \langle a, d, b, e, f \rangle$ ,  $\langle n_2, n_6, n_4, n_7, n_8, n_{11}, n_{13}, n_{12}, n_{14} \rangle \equiv \langle a, c, b, d, e, f, h, g \rangle$ , etc.

## 2.2. Alineamientos

Los alineamientos relacionan eventos de una traza con las actividades de un modelo de proceso, en nuestro caso, las hojas de un árbol de proceso. A lo largo del artículo, se utilizará  $A$  para representar las actividades de la traza, y  $N$  para representar las hojas de un árbol de proceso  $P$ . Adicionalmente, utilizamos  $\gg$ , (i.e.,  $\gg \notin A, \gg \notin N$ ) para representar *saltos*. Así, un alineamiento se representa como una secuencia  $(A \gg \times N \gg)^*$ . Cada par  $(a, n) \in (A \gg \times N \gg)$  es considerado como un *movimiento*. En un alineamiento se pueden distinguir los siguientes movimientos  $(a, n)$ : *i*) movimiento síncrono, si  $a \in A$  y  $n \in N$  tal que la etiqueta de  $n$  es igual a la etiqueta de  $a$  o  $a = \gg$  y  $n \in N$  tal que la etiqueta de  $n$  es  $\tau$ ; *ii*) movimiento en el modelo, si  $a = \gg$  y  $n \in N$ ; y *iii*) movimiento en la traza, si  $a \in A$  y  $n = \gg$ . Otro tipo de movimientos se consideran *ilegales*. Así, dada una traza  $\sigma$  y un árbol de proceso  $P$ , a lo largo del artículo se utilizará la notación  $\gamma_{(\sigma, P)}$  para denotar un alineamiento entre  $\sigma$  y  $P$ .

Por ejemplo, considerando la traza  $\sigma_1 = \langle a, b, c, e, f \rangle$ , y la secuencia de hojas  $\langle n_2, n_4, n_6, n_8, n_{11}, n_{14} \rangle \equiv \langle a, b, c, e, f \rangle$  del árbol de proceso  $P_1$  de la Figura 1, es posible relacionar  $\sigma_1(1)$  con  $n_2$ ,  $\sigma_1(2)$  con  $n_4$ , ...,  $\sigma_1(5)$  con  $n_{11}$ , generando el siguiente alineamiento<sup>3</sup>:

$$\gamma_{(\sigma_1, P_1)}^1 = \frac{A \gg}{N \gg} \left\| \begin{array}{c|c|c|c|c|c} a & b & c & e & f & \gg \\ \hline n_2 & n_4 & n_6 & n_8 & n_{11} & n_{14} \end{array} \right\|$$

Este alineamiento  $\gamma_{(\sigma_1, P_1)}^1$  no es la única posible relación de  $\sigma_1$  con  $P_1$ . Otra posible combinación sería:

$$\gamma_{(\sigma_1, P_1)}^2 = \frac{A \gg}{N \gg} \left\| \begin{array}{c|c|c|c|c|c|c|c|c|c} a & b & c & e & f & \gg & \gg & \gg & \gg & \gg \\ \hline n_2 & n_4 & n_6 & n_8 & n_{11} & n_{13} & n_{12} & n_{14} & & \end{array} \right\|$$

<sup>3</sup>Hay que tener en cuenta que  $n_{14}$  es una hoja con la etiqueta  $\tau$  y por lo tanto se puede crear un movimiento síncrono  $(\gg, n_{14})$ .

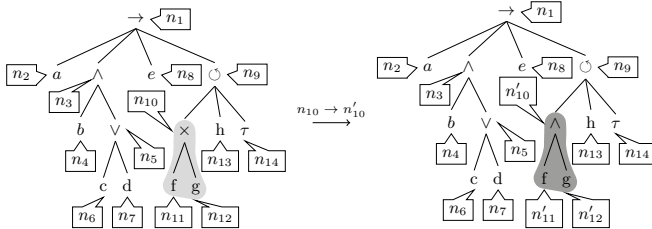
En general, dado un modelo de proceso y una traza, existen una gran multitud de posibles alineamientos. Sin embargo, nuestro interés reside en obtener un alineamiento óptimo. Un alineamiento óptimo es un alineamiento que minimiza una función coste de tipo  $\kappa : (A^{\gg} \times N^{\gg})^* \rightarrow \mathbb{R}^+$ . Generalmente, un movimiento síncrono suele tener asignado un coste de 0, mientras que un movimiento ilegal tiene asignado un coste  $\infty$ . Por otro lado, el coste de los movimientos en modelo/traza suelen ser dependientes del dominio, pero generalmente mayores que 0. En esencia, un alineamiento óptimo busca minimizar el número de movimientos en el modelo y en las trazas. De este modo, considerando esta función de coste,  $\gamma_{(\sigma_1, P_1)}^1$  es mejor que  $\gamma_{(\sigma_1, P_1)}^2$  ya que tiene un menor número de movimientos de tipo  $(\gg, n)$ . La principal asunción de este artículo es que existe una función *oráculo*  $f$  que, dada una traza  $\sigma$  y un modelo  $P$ , produce un alineamiento (óptimo). Actualmente, es posible obtener estos alineamientos óptimos para una red de Petri o un árbol de proceso a partir de, por ejemplo, el algoritmo  $A^*$  [10].

### 3. Reparación de alineamientos

Como se ha indicado, algunas técnicas de minería de procesos comparten una propiedad interesante, es decir, es necesario calcular alineamientos para varios modelos de proceso relativamente similares. En adelante, la hipótesis que se plantea en este artículo se puede definir del siguiente modo. Dada una traza  $\sigma_L$ , los árboles de proceso  $P$  y  $P'$ , y un alineamiento  $\gamma_{(\sigma_L, P)}$ , ¿es posible calcular  $\gamma_{(\sigma_L, P')}$  reutilizando y reparando  $\gamma_{(\sigma_L, P)}$ ?

#### 3.1. Ejemplo concreto

En este artículo se ilustra la reparación de un alineamiento a partir de un ejemplo de ejecución del propio algoritmo. Para ello, se utilizarán los árboles de proceso  $P_1$  y  $P_2$  (Figure 2) como punto de partida. Además, consideraremos la



**Figura 2.** Modificación del nodo  $n_{10}$  en el árbol de proceso  $P_1$  (izquierda), dando como resultado el árbol de proceso  $P_2$  (derecha). La rama de cambio  $S_1$  está resaltada en el árbol de proceso  $P_1$  con gris claro, y la rama  $S_2$  está resaltada en  $P_2$  con gris oscuro.

traza  $\sigma_3 = \langle a, c, d, e, f, g, h, g, f \rangle$  y el alineamiento  $\gamma_{(\sigma_3, P_1)}$ :

$$\gamma_{(\sigma_3, P_1)} = \frac{A \gg}{N \gg} \left| \begin{array}{c|c|c|c|c|c|c|c|c|c|} a & \gg & c & d & e & f & g & h & g & f & \gg \\ \hline n_2 & n_4 & n_6 & n_7 & n_8 & n_{11} & \gg & n_{13} & n_{12} & \gg & n_{14} \end{array} \right|$$

Como se puede apreciar, a la traza  $\sigma_3$  le falta la actividad  $b$ , impuesta por el nodo  $n_3$  ( $\wedge$ ), dando lugar al movimiento en el modelo ( $\gg, n_4$ ). Es más,  $n_{10}(\times)$  no permite la ejecución de las actividades  $f$  y  $g$  sin que  $h$  aparezca entre ellas, dando lugar a dos movimientos en la traza ( $g, \gg$ ) y ( $f, \gg$ ). Esto se puede solucionar cambiando el operador del nodo  $n_{10}$  ya sea al operador  $\wedge$  o  $\vee$ . El árbol de proceso  $P_2$  que se muestra en la parte de la derecha de la Figura 2 muestra el cambio del operador del nodo  $n_{10}$  al operador  $\wedge$ . Así, el proceso para calcular  $\gamma_{(\sigma_3, P_2)}$  reutilizando  $\gamma_{(\sigma_3, P_1)}$  consiste en tres etapas: *i*) detección de la rama de cambio, *ii*) cálculo de los alineamientos, y *iii*) reconstrucción del alineamiento.

*Etapas 1; Detección de la rama de cambio* El primer paso en reusar  $\gamma_{(\sigma_3, P_1)}$  implica detectar las diferencias de  $P_1$  con respecto a  $P_2$  y vice versa, es decir, las ramas de cambio entre los dos procesos. En la Figura 2 aparecen resaltadas las ramas de cambio  $S_1$  y  $S_2$  en gris claro y gris oscuro, respectivamente.  $S_1$  consiste en los nodos  $n_{10}$ ,  $n_{11}$  y  $n_{12}$  de  $P_1$ , mientras que  $S_2$  consiste en los nodos  $n'_{10}$ ,  $n'_{11}$  y  $n'_{12}$  de  $P_2$ . Como se puede apreciar, estas dos ramas son a su vez dos árboles de proceso, donde, por ejemplo, para  $S_1$  el nodo raíz es  $n_{10}$ . Utilizando esta información, es necesario detectar qué movimientos de  $\gamma_{(\sigma_3, P_1)}$  pertenecen a la rama de cambio de  $P_1$ , es decir, que movimientos están relacionados con los nodos de  $S_1$ .

Utilizando estas reglas, es posible empezar a construir el nuevo alineamiento  $\gamma_{(\sigma_3, P_2)}$ . Cada movimiento  $(a, n) \in \gamma_{(\sigma_3, P_1)}$  no perteneciente a  $S_1$ , permanece intacto y se copia en la misma posición exacta en el nuevo alineamiento  $\gamma_{(\sigma_3, P_2)}$ . Por otro lado, si  $(a, n)$  pertenece a  $S_1$ , este no se copia en  $\gamma_{(\sigma_3, P_2)}$ , aunque para cada uno de estos movimientos recordamos su posición exacta en  $\gamma_{(\sigma_3, P_1)}$ . Es más, cada vez que sucede el primer movimiento  $(a, n)$  dentro de la rama de cambio  $S_1$ , se crea una secuencia intermedia  $\sigma_3^1 = \langle a \rangle$  (o  $\sigma_3^1 = \epsilon$  si  $a = \gg$ ). Por cada movimiento  $(a', n')$  subsecuente que pertenece a la rama de cambio  $S_1$ , se actualiza  $\sigma_3^1$  a  $\sigma_3^1 \cdot \langle a' \rangle$  (o  $\sigma_3^1 \cdot \epsilon = \sigma_3^1$  si  $a' = \gg$ ). Sin embargo, si durante este proceso se detecta un movimiento  $(a'', n'')$  perteneciente a  $S_1$  que indica que es una nueva ejecución de la rama de cambio  $S_1$ , entonces se crea una nueva traza  $\sigma_3^2 = \langle a'' \rangle$  (o  $\sigma_3^2 = \epsilon$  si  $a'' = \gg$ ). Remarcar que este tipo de comportamiento puede estar presente en un alineamiento debido a construcciones de tipo lazo en el modelo de proceso. Esta primera etapa, aplicada al ejemplo introducido al principio de la sección, se describe en la Tabla 1. Los primeros cinco elementos de  $\gamma_{(\sigma_3, P_1)}$  están fueran de la rama de cambio  $S_1$ , por lo tanto se copian directamente en  $\gamma_{(\sigma_3, P_2)}$ . El sexto y séptimo movimiento, es decir,  $(f, n_{11})$  y  $(g, \gg)$ , pertenece a la rama de cambio  $S_1$ , por lo tanto no se copian, pero sí se guarda su posición original. Adicionalmente se crea  $\sigma_3^1$  a partir de estos dos movimientos. El octavo movimiento vuelve a estar fuera de la rama de cambio  $S_1$ , y se vuelve a copiar directamente en  $\gamma_{(\sigma_3, P_2)}$ . El noveno y décimo movimiento, es decir  $(g, n_{12})$  y  $(f, \gg)$ , vuelven a estar de nuevo dentro de la rama de cambio  $S_1$ . Sin embargo, estos

**Tabla 1.** Visión general de la primera etapa del algoritmo de reparación de alineamientos.

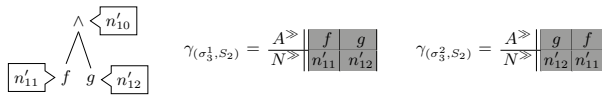
$A \gg$	$a$	$\gg$	$c$	$d$	$e$	$f$	$g$	$h$	$g$	$f$	$\gg$
$N \gg$	$n_2$	$n_4$	$n_6$	$n_7$	$n_8$	$n_{11}$	$\gg$	$n_{13}$	$n_{12}$	$\gg$	$n_{14}$
	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	⏟		$\downarrow$	⏟		$\downarrow$
$A \gg$	$a$	$\gg$	$c$	$d$	$e$	$\sigma_3^1 = (f, g)$		$h$	$\sigma_3^2 = (g, f)$		$\gg$
$N \gg$	$n_2$	$n_4$	$n_6$	$n_7$	$n_8$	$\sigma_3^1 = (f, g)$		$n_{13}$	$\sigma_3^2 = (g, f)$		$n_{14}$

dos movimientos indican una nueva ejecución de la rama de cambio  $S_1$  y, por lo tanto, se debe de crear una nueva traza  $\sigma_3^2$  a partir de estos dos elementos. Finalmente el último movimiento del alineamiento está fuera de la rama de cambio de  $S_1$ . En este paso, se recorren linealmente todos los movimientos de  $\gamma_{(\sigma_3, P_1)}$ , comprobando para cada  $(a, n) \in \gamma_{(\sigma_3, P_1)}$  si pertenece o no a la rama de cambio  $S_1$ . Para determinar si un movimiento  $(a, n)$  pertenece a la rama de cambio, se definen las siguientes dos reglas:

1. Si  $n$  es una hoja de  $S_1$ ,  $(a, n)$  pertenece a  $S_1$ .
2. Si  $n = \gg$ , y el movimiento previo  $(a', n') \in \gamma_{(\sigma_3, P_1)}$  pertenece a  $S_1$ , entonces  $(a, n)$  también pertenece a  $S_1$ .

*Etap 2; Cálculo de los alineamientos* La etapa anterior tiene como salida el alineamiento parcialmente completo  $\gamma_{(\sigma_3, P_2)}$ , así como una serie de subtrazas, es decir, la parte inferior de la Tabla 1. El siguiente paso del proceso de reparación consiste en crear nuevas porciones de alineamientos para estas subtrazas generadas a partir de los elementos que pertenecen a  $S_1$ . Como la rama de cambio  $S_2$  es en sí un árbol de proceso, es posible utilizar la función *oráculo*  $f$  para calcular los dos nuevos alineamientos  $\gamma_{(\sigma_3^1, S_2)}$  y  $\gamma_{(\sigma_3^2, S_2)}$ . El resultado de este cálculo, junto con la rama de cambio  $S_2$ , se muestra en la Figura 3.

*Etap 3; Reconstrucción del alineamiento* El último paso implica reconstruir el alineamiento a partir de las porciones creadas en la *etapa 2* y el alineamiento inferior de la Tabla 1. Durante las etapas previas, se ha guardado la posición de cada movimiento en  $\gamma_{(\sigma_3, P_1)}$  que pertenecía a  $S_1$ . Por lo tanto, por cada movimiento  $(a, n)$  dentro de la rama de cambio en  $\gamma_{(\sigma_3, P_1)}$  con  $a \neq \gg$ , hay un movimiento  $(a, n')$  ya sea en  $\gamma_{(\sigma_3^1, S_2)}$  o  $\gamma_{(\sigma_3^2, S_2)}$ . En el presente ejemplo,  $(f, n_{11})$  vs.  $(f, n'_{11})$  en  $\gamma_{(\sigma_3^1, S_2)}$ ,  $(g, \gg)$  vs.  $(g, n'_{12})$  en  $\gamma_{(\sigma_3^1, S_2)}$ ,  $(g, n_{12})$  vs.  $(g, n'_{12})$  en  $\gamma_{(\sigma_3^2, S_2)}$



**Figura 3.** Árbol de proceso  $S_2$  y los dos alineamientos  $\gamma_{(\sigma_3^1, S_2)}$  y  $\gamma_{(\sigma_3^2, S_2)}$ .

Reparación de alineamientos en modelos de proceso similares 7

**Tabla 2.** Visión general de la tercera etapa del algoritmo de reparación de alineamientos.

$A \gg$	$a$	$\gg$	$c$	$d$	$e$	$\sigma_3^1 = (f, g)$	$h$	$\sigma_3^2 = (g, f)$	$\gg$		
$N \gg$	$n_2$	$n_4$	$n_6$	$n_7$	$n_8$		$n_{13}$		$n_{14}$		
	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$		$\downarrow$	$\downarrow$	$\downarrow$		
$A \gg$	$a$	$\gg$	$c$	$d$	$e$	$f$	$g$	$h$	$f$	$\gg$	
$N \gg$	$n_2$	$n_4$	$n_6$	$n_7$	$n_8$	$n_{11}$	$n_{12}$	$n_{13}$	$n_{12}$	$n_{11}$	$n_{14}$

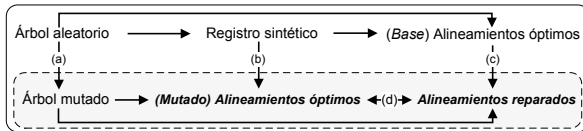
y  $(f, \gg)$  vs.  $(f, n'_{11})$  en  $\gamma_{(\sigma_3^2, S_2)}$ . Este tipo de movimientos sirven como *puntos de ancla* para reubicar los nuevos fragmentos de alineamiento en el parcialmente finalizado alineamiento de la *etapa 1*. Una vez realizado el tercer paso, la salida es un alineamiento de  $\sigma_3$  sobre  $P_2$ , sin necesidad de calcular un alineamiento sobre todo  $P_2$  desde cero. Este último paso, mostrando el alineamiento  $\gamma_{(\sigma_3, P_2)}$ , está descrito en la Tabla 2.

### 4. Evaluación

En esta sección *i)* se compara el tiempo necesario para reparar un alineamiento frente a calcularlo desde cero, y *ii)* se mide la calidad de los nuevos alineamientos con respecto al óptimo, es decir, el calculado desde cero.

#### 4.1. Configuración de la experimentación

La Figura 4 muestra el esquema seguido durante la experimentación. En una primera fase, generamos un árbol de proceso aleatorio con un tamaño aleatorio. Utilizando este modelo como base, generamos un registro de eventos, con ruido, de 2000 trazas. A continuación, calculamos los alineamientos óptimos de todas las trazas en el registro de eventos con respecto al modelo base inicial. En una segunda fase, realizamos una serie de cambios, e.g. eliminar, añadir, o modificar partes del modelo de proceso, sobre el modelo base inicial (paso b en la Figura 4), generando un total de 150 modelos diferentes. Luego, por cada modelo modificado, calculamos dos tipos de alineamientos, los óptimos a partir únicamente del



**Figura 4.** Proceso seguido durante la experimentación.

registro de eventos (paso b en la Figura 4), y los reparados reutilizando los alineamientos óptimos previamente calculados (paso c en la Figura 4). Finalmente comparamos ambos alineamientos (paso d en la Figura 4).

Siguiendo este proceso, se han creado un total de 25 árboles de proceso iniciales con un tamaño aleatorio de entre 20 y 31 nodos, incluyendo tanto las actividades (hojas) como los operadores (resto de nodos). Por lo tanto, la presente técnica se ha validado sobre  $25 \times 150 \times 2000 \approx 7,5 \cdot 10^6$  alineamientos diferentes.

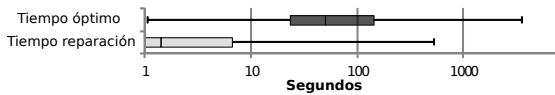
#### 4.2. Tiempo de cálculo

Como se puede apreciar en la Figura 5, no existe solapamiento alguno entre el segundo y tercer cuartil al comparar ambos tiempos. Esto quiere decir que en casi todos los casos, el tiempo necesario para alinear un modelo y un registro de eventos a través de la reparación de alineamientos es inferior al tiempo necesario para calcular los alineamientos. En este caso, el tiempo necesario para reparar un alineamiento está directamente relacionado con el tamaño de la rama de cambio, lo que explica el largo tamaño del *bigote* derecho al reparar un alineamiento. Claramente, si el punto de cambio es el nodo raíz del árbol de proceso, la rama de cambio será el árbol de proceso al completo. Por lo tanto, el tiempo necesario para aplicar esta técnica será ligeramente similar a calcular el alineamiento óptimo desde cero. Dados estos resultados, podemos concluir que, utilizando la técnica presentada en este artículo, se garantiza un tiempo mucho menor o, en el peor de los casos igual, que calcular un nuevo alineamiento desde cero.

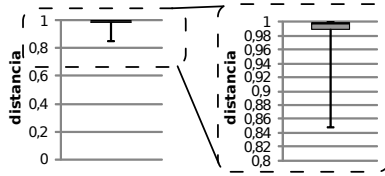
#### 4.3. Calidad de los alineamientos

Una de las principales limitaciones de esta técnica es que no se puede garantizar que el alineamiento resultante tras la reparación sea óptimo. Sin embargo, medir *lo bueno* que es un alineamiento reparado no es una tarea sencilla, ya que sería necesario recorrer todos los posibles alineamientos entre un modelo de proceso y un registro de eventos, lo que es computacionalmente inviable en esta experimentación.

Por lo tanto, para poder medir la calidad de un alineamiento reparado, proponemos una medida de *distancia*, que mide la distancia de un alineamiento



**Figura 5.** Diagramas de caja, en escala logarítmica, mostrando el tiempo necesario para reparar un alineamiento respecto a calcular el alineamiento óptimo desde cero.



**Figura 6.** Diagrama de caja mostrando las distancias respecto al alineamiento óptimo de los alineamientos reparados.

reparado con respecto al alineamiento óptimo. Para calcular esta distancia, primero calculamos el coste del alineamiento óptimo  $\gamma^*$ . Adicionalmente, creamos un alineamiento  $\gamma^w$ , denominado *el mejor de los peores alineamientos*, que consiste únicamente de movimientos en la traza ( $a, \gg$ ) y movimientos en el modelo ( $\gg, n$ ). Finalmente, calculamos el coste del alineamiento reparado  $\gamma^r$ . Así, sean  $c^*$ ,  $c^w$  y  $c^r$  el coste para  $\gamma^*$ ,  $\gamma^w$  y  $\gamma^r$ , respectivamente, la *distancia* del alineamiento será  $distancia(\gamma^r) = 1 - \frac{c^r - c^*}{c^w - c^*}$ . Por lo tanto, si  $distancia(\gamma^r) = 1$ ,  $\gamma^r$  será igual al alineamiento óptimo  $\gamma^*$ .

La Figura 6 muestra el diagrama de caja para la *distancia* media calculada para los alineamientos reparados. Tal y como se puede apreciar, siempre se obtiene una *distancia* superior al 0.84, y en el 75 % de todos los experimentos está por encima del 0.98. Por lo tanto, en los casos en los que, tras la reparación, no se obtiene un alineamiento óptimo, la diferencia del alineamiento reparado respecto al óptimo es mínima. Es decir, la pérdida de optimalidad está limitada y se encuentra entre límites aceptables. De nuevo, existe una relación directa entre el tamaño de la rama de cambio y la potencial pérdida de optimalidad. Si el punto de cambio está próximo al nodo raíz del árbol de proceso, se incluirán menos movimientos del alineamiento original en el nuevo alineamiento. En consecuencia, la probabilidad de obtener un alineamiento óptimo será mayor. Por lo tanto, si el punto de cambio es el nodo raíz, podemos garantizar optimalidad en todos los casos, aunque esto implicaría una ganancia nula de tiempo, ya que implicaría recalcular el alignment al completo.

## 5. Conclusiones

En este artículo hemos presentado una nueva técnica para calcular alineamientos a partir de alineamientos ya existentes, sin necesidad de calcular los nuevos alineamientos desde cero. Los resultados obtenidos con un conjunto de registros y modelos de procesos aleatorios muestran que el tiempo de aplicar esta solución siempre es significativamente mejor, o igual, que calcular los nuevos alineamientos desde cero. Además, los resultados obtenidos muestran que

10 B. Vázquez-Barreiros et al.

la potencial pérdida de optimalidad es limitada y se encuentra entre límites aceptables.

**Agradecimientos.** Este trabajo ha sido financiado con fondos del Ministerio de Economía y Competitividad (proyecto TIN2014-56633-C3-1-R, cofinanciado por el Fondo Europeo de Desarrollo Regional - FEDER ), el Ministerio de Galicia y Educación bajo los proyectos EM2014/012, CN2012/151, GRC2014/030, y el programa de investigación DELIBIDA con el apoyo de NWO.

## Referencias

1. Aalst, W.M.P.v.d., Adriansyah, A., Dongen, B.F.v.: *Replaying History on Process Models for Conformance Checking and Performance Analysis*. Wiley Interdisc. Rev.: *Data Mining and Knowledge Discovery* **2**(2) (2012) 182–192
2. Adriansyah, A., Munoz-Gama, J., Carmona, J., Dongen, B.F.v., Aalst, W.M.P.v.d.: *Measuring Precision of Modeled Behavior*. *Inf. Syst. E-Business Management* **13**(1) (2015) 37–67
3. Buijs, J.C.A.M.: *Flexible Evolutionary Algorithms for Mining Structured Process Models*. PhD thesis, Eindhoven University of Technology (2014)
4. Leemans, S.J.J., Fahland, D., Aalst, W.M.P.v.d.: *Process and Deviation Exploration with Inductive Visual Miner*. In: *Proceedings of the BPM Demo Sessions 2014*, Eindhoven, The Netherlands, September 10, 2014. (2014) 46
5. Eck, M.L.v., Buijs, J.C.A.M., Dongen, B.F.v.: *Genetic Process Mining: Alignment-Based Process Model Mutation*. In: *Business Process Management Workshops - BPM 2014 International Workshops*, Eindhoven, The Netherlands, September 7-8, 2014, Revised Papers. (2014) 291–303
6. Aalst, W.M.P.v.d.: *Decomposing Petri Nets for Process Mining: A Generic Approach*. *Distributed and Parallel Databases* **31**(4) (2013) 471–507
7. Leoni, M.d., Munoz-Gama, J., Carmona, J., Aalst, W.M.P.v.d.: *Decomposing Alignment-Based Conformance Checking of Data-Aware Process Models*. In: *OTM 2014*, Amantea, Italy, October 27-31, 2014, Proceedings. (2014) 3–20
8. Munoz-Gama, J., Carmona, J., Aalst, W.M.P.v.d.: *Single-Entry Single-Exit Decomposed Conformance Checking*. *Inf. Syst.* **46** (2014) 102–122
9. Leemans, S.J.J., Fahland, D., Aalst, W.M.P.v.d.: *Discovering block-structured process models from event logs - A constructive approach*. In: *Petri Nets 2013*, Milan, Italy, June 24-28. Proceedings. (2013) 311–329
10. Adriansyah, A.: *Aligning Observed and Modeled Behavior*. PhD thesis, Eindhoven University of Technology (2014)



# Defining PPIs for Process Variants based on Change Patterns\*

Bedilia Estrada-Torres<sup>1</sup>, Victoria Torres<sup>2</sup>, Adela del-Río-Ortega<sup>1</sup>, Manuel Resinas<sup>1</sup>, Vicente Pelechano<sup>2</sup> and Antonio Ruiz-Cortés<sup>1</sup>

<sup>1</sup> Depto. de Lenguajes y Sistemas Informáticos, Universidad de Sevilla, Sevilla, Spain.

<sup>2</sup> Centro de Investigación en Métodos de Producción de Software, Universitat Politècnica de València, València, Spain.

**Abstract.** Business Process (BP) families are made up of BP variants that share commonalities but also show differences to accommodate the specific necessities of different application contexts (i.e., country regulations, industrial domain, etc.). Even though there are modelling techniques to represent these families (e.g., C-EPC, Provop), there is no work aimed at the performance measurement of the different BP variants that conform the family. Process Performance Indicators (PPI) are commonly used to study and analyse the performance of business processes. However, the application of such indicators in BP families increases the modelling and management complexity of the whole family. To deal with this complexity, this work introduces a modelling solution for managing PPI variability based on the concepts of change patterns for process families (CP4PF). The proposed solution includes a set of patterns aimed at 1) reducing the number of operations required to specify PPIs and 2) ensuring PPI family correctness.

## 1 Introduction

The increasing adoption of Process-Aware Information Systems (PAISs) during the last decade has resulted in large process model repositories which usually comprise collections of related process model variants (process variants (PVs) for short). While PVs pursue the same or similar business objectives (e.g., good delivery), they show differences depending on their application context (e.g., country regulations). This collection of PVs defines a Business Process (BP) family.

Managing PVs is a complex task that requires the use of specific approaches, such as C-EPC [1] or PROVOP [2]. Nevertheless, there is no work aimed at the performance measurement of the different PVs.

Process Performance Indicators (PPI) are commonly used to study and analyse the performance of business processes. However, the application of such indicators in BP families increases the modelling and management complexity of

\* This work was partially supported by the European Commission (FEDER), the Spanish and the Andalusian R&D&I programmes (grants TIN2015-70560-R (BELI), P12-TIC-1867 (COPAS), P10-TIC-5906 (THEOS) and TIN2013-42981-P (SMART-ADAPT)), and the Spanish network TIN2014-53986-REDT (RCIS).

the whole family and may require the definition of new mechanisms that help during their definition and subsequent management.

To address this complexity, this paper proposes investigating the application of change patterns for process families (CP4PF) [3] to the definition of PPIs in PVs. The main reason to conduct the research based on Change Patterns (CPs) is that these may reduce the number of operations required to specify PPIs and ensure the correctness of PPIs along the whole BP family.

## 2 Change Patterns for PPIs in Process Variants

CP4PF allow the modeling and evolution of the process families, which are intended to reduce the effort needed for such purposes [3]. On that basis we seek to define a set of general CPs for managing PPI variants. The following steps are deemed necessary to the implementation of these CPs:

**Define modelling restrictions:** First, it is necessary to define a set of restrictions allowing the correct definition and configuration of PPIs. Restrictions may be constructed considering specification of languages for PPI definitions.

**Design change patterns:** Bearing in mind CP4PF structures described in [3] and PPI variability criteria introduced in previous work of some of the authors [4], we have identified a preliminary set of operations to be implemented for configuring PPI variants. Those operations are shown below:

Let  $\mathcal{BBP}$  be a base business process (process model where context-specific structural adaptations are applied in order to derive a particular PV) where a set of PPIs are defined as a PPI variant.

- **INSERT<sub>P</sub>**: A PPI is inserted when it is not defined in the  $\mathcal{BBP}$ , but it is required in the PV being derived.
- **INSERT<sub>M</sub>**: A measure is inserted when the structure of a PPI needs to be changed in the PV being derived.
- **DELETE<sub>P</sub>**: A PPI is deleted when it is defined in the  $\mathcal{BBP}$ , but it is not required in the PV being derived.
- **DELETE<sub>M</sub>**: A measure is deleted when the structure of a PPI needs to be changed in the PV being derived.
- **MODIFY<sub>P</sub>**: A PPI attribute is modified (e.g., scope) when in the  $\mathcal{BBP}$  it is specified in a different way from the required in the PV being derived.
- **MODIFY<sub>M</sub>**: A measure attribute is modified when the requirements of the measure in the  $\mathcal{BBP}$  are different from those required in the PV being derived (Changes depends on the type of measure).
- **MOVE<sub>P</sub>**: A PPI is moved when it is defined in the  $\mathcal{BBP}$  over a set of BP elements, but those BP elements are different in the PV being derived.

Then, each operation is represented by means of CPs for PPIs variants and each CP should be constructed or adapted for fulfilling the set of restrictions defined in the first step. The configuration of PPI variants should be made by means of *Adjustment Points*, where we define the starting and ending points of each operation. Depending on the restrictions previously defined, it is possible to specify more than one CP for each operation (e.g.,  $INSERT_M$ , is related to as many CPs as type of measures exist in restrictions).

**Instantiate change patterns:** In this step, it is necessary to choose a language for modelling BP families and a language for PPI definitions that will be used as a basis for applying CPs.

**Validate change patterns:** In order to validate the CPs proposed, a case study must be selected that should reflect variability in PPI definitions. For performing this step, we will follow different validation techniques, such as (i) experiments performed by real users for evaluating usability and complexity of CPs, and (ii) count and compare operations required in the modelling and configuration of PPIs using CPs and without them, in order to validate whether the amount of operations is reduced when CPs are used.

**Implement CPs:** Finally, a software tool that supports the modelling and management of PPIs in PVs would be of utmost usefulness for final users.

### 3 Example of Change Pattern

Let us suppose a Process Family (PF) associated with performing customer-facing order management and order fulfillment activities. The PF is comprised of two PVs: the intention of PV-1 is to have the product or service available when a customer order arrives, whilst PV-2 configures, manufactures, and/or assembles the product from standard raw materials, parts or ingredients, in response to a specific firm customer order.

Although the control-flow of both PVs are different, they also shows similarities, e.g. both PVs includes the same PPI to measure a specific activity in each variant. Just as there are BPMLs focused on the reuse of some common parts of PVs, we propose to reuse PPI definitions (or some parts of them).

Figure 1 shows a PPI defined for the PV-1, which measures the execution time of the activity *Load Vehicle and Generate Shipping Documents - LV&GSD*. The same PPI is defined for PV-2, this time to measure the activity *Load Product and Generate Shipping Docs - LP&GSD*. For reusing this PPI definition we need to first disconnect the PPI from PV-1 and then to reconnect it to PV-2. This change in PVs results in the application of 4 actions: 2 *delete actions* for deleting connections and 2 *add actions* to reconnect the measure with the PV.

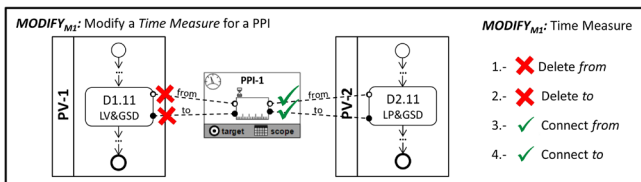


Fig. 1. Example of actions required for an operation  $MODIFY_M$  (Time Measure).

In order to reduce the number of required operations and also to ensure the changes are performed correctly, we propose to use CPs for PPI definitions. For our example we should define a CP for the operation  $MODIFY_M$ . This operation can be related to as many CP as type of measures exist in the restrictions. This example is associated with a time measure.

The application of each CP is considered as a single action that requires a set of parameters to specify the internal actions to be carried out.

The example of PPI in Figure 1 has only one measure that needs to be reconnected, but when the PPI has more than one measure, CPs are more meaningful because the amount of actions for each operation is reduced significantly.

## 4 Ongoing and Future Work

The technique used for the management of PPI variants is being currently developed and applied in a case study based on processes and metrics of SCOR [5]. We use *PROVOP* for the modelling of BPs and their PVs involved. Moreover, *PPINOT* [6] and variability restrictions defined in [4] are used to model PPIs and to configure their variants.

In order to validate our technique, we also plan to apply it to the process of organ donation and transplantation, based on information provided by surgeons from various Spanish hospitals. The validation techniques will be similar to those used in [3], performing experiments with real users and counting and comparing operations required in each definition and configuration of PPIs, for concluding whether or not the number of operations is reduced.

Finally, we plan to implement a software tool that assists users to manage variability in the control-flow and performance perspective of BPs by using CPs.

## References

1. Rosemann, M., van der Aalst, W.M.P.: A configurable reference modelling language. *Information Systems* **32**(1) (2007) 1–23
2. Hallerbach, A., Bauer, T., Reichert, M.: Capturing variability in business process models: the provop approach. *J. of Software Maintenance* **22**(6-7) (2010) 519–546
3. Ayora, C., Torres, V., de la Vara, J.L., Pelechano, V.: Variability management in process families through change patterns. *Inform Software Tech* **74** (2016) 86 – 104
4. Estrada-Torres, B., del-Río-Ortega, A., Resinas, M., Ruiz-Cortés, A.: Identifying Variability in Process Performance Indicators. In: *Business Process Management Forum 2016. Lecture Notes in Business Information Processing* (in press)
5. APICS, S.C.C.: Supply Chain Operations Reference Model: SCOR Version 11.0. Supply Chain Council (2015) APICS, CCOR, CPIM, CSCP, DCOR, SCOR, and SCORmark are all registered trademarks of APICS. All rights reserved.
6. del Río-Ortega, A., Resinas, M., Cabanillas, C., Ruiz-Cortés, A.: On the definition and design-time analysis of process performance indicators. *Information Systems* **38**(4) (2013) 470–490

# Incertidumbre de Datos en el Modelado de Procesos de Negocio

Priscill Orue, Carmen Morcillo, and Antonio Vallecillo

Universidad de Málaga, Spain.  
{priscill.orue, aixela, av}@uma.es

**Resumen** Este artículo describe nuestro trabajo en curso sobre el modelado de procesos de negocio teniendo en cuenta la incertidumbre de los datos. Esto es un aspecto esencial para modelar con mayor precisión y fidelidad procesos industriales en donde ciertos parámetros –como la duración de las tareas o el coste de algunos recursos– no pueden determinarse con exactitud, pero son necesarios tenerlos en cuenta a la hora de analizar los sistemas.

## 1. Introducción

Los modelos actuales de software, tales como BPMN o UML, trabajan con valores exactos para la representación de las propiedades de un sistema, algo que limita su utilidad para el trabajo con sistemas realistas que presenten propiedades de incertidumbre [1]. Un dominio relevante de aplicación es el modelado de procesos de negocio, en el que la duración de las tareas y los costes de los recursos son usualmente difíciles de estimar de forma precisa. La incertidumbre puede deberse a datos de entrada inconsistentes, parámetros con errores numéricos, por desconocimiento del comportamiento del sistema o por la naturaleza tácita del conocimiento humano [2] [3].

En este trabajo, proponemos la extensión de UML y OCL para incorporar la incertidumbre de los datos a modelos software. Hemos definido el tipo `UncertainReal`, como una superclase del tipo `UML Real`, con un conjunto de operaciones para los valores de este tipo. Se proporcionan expresiones analíticas para implementar estas operaciones, así como una librería Java [4]. Primeramente, describimos los antecedentes sobre la incertidumbre de datos. A continuación, detallamos la incorporación de la incertidumbre al modelado de procesos de negocios. Finalmente, presentamos las conclusiones del trabajo y delineamos el trabajo futuro con respecto a la incertidumbre de datos.

## 2. Antecedentes

La incertidumbre en la medida se define en ISO VIM [5] como un parámetro asociado al resultado de una medida, que caracteriza la dispersión de los valores de un atributo medible. Para los propósitos de este trabajo, se considera la incertidumbre de riesgo, la cual es cuando se conocen todos los posibles valores de un parámetro [6].

Supongamos una línea de montaje de una fábrica organizado en procesos de manufacturación. Cada proceso está formado por tareas (*task*) que tienen entradas (recursos utilizados) y producen salidas, con una duración y coste asociados, combinables en forma secuencial o paralela (Fig. 1). Si se representa la duración  $d$  del ensamblado de cierto componente con un rango entre 2 y 3 seg (según la persona que lleva a cabo el

paso) o que el coste  $c$  de una parte puede variar entre 1.30 y 1.36 euros (según la tasa de cambio diario del dólar), los modelos de software actuales tienden a usar los valores promedio, asignando  $d = 2.5$  seg. al paso y  $c = 1.33$  euros a la parte producida. Sin embargo, sería más expresivo representar estas cantidades por medio de intervalos  $d = [2.0, 3.0]$  o  $c = [1.30, 1.36]$ .

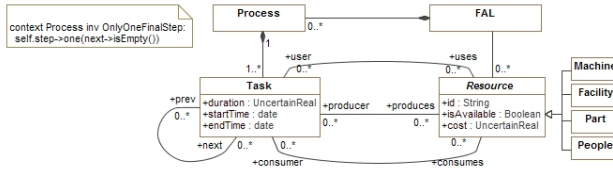


Figura 1. Metamodelo simple para la especificación de procesos

Para representar la incertidumbre se puede simular la coordinación de flujos de materiales [7] o fijar niveles de incertidumbre [8]. No obstante, los resultados obtenidos no son generalizables. Otra posibilidad es considerar las especificaciones declarativas de alto nivel para generar modelos configurables de procesos de negocios optimizados, y asignar a los datos con incertidumbre distribuciones de probabilidad, como se hace en [3]. Sin embargo, este enfoque tiene limitaciones debido a que obliga a combinar las correspondientes distribuciones de probabilidad cuando se combinan datos con incertidumbre. Esto se conoce como propagación de incertidumbre y en general no admite soluciones analíticas, sino mediante simulaciones [2].

### 3. Datos Inciertos en el Modelado de Procesos de Negocio

La *Guía para la Expresión de la Incertidumbre en Mediciones (GUM)* [2] define el término incertidumbre estándar como la incertidumbre del resultado de una medida expresada como una desviación estándar. A partir de esta definición, proponemos una extensión de UML para declarar variables de incertidumbre por medio de un nuevo tipo llamado *UncertainReal* (Fig. 2), el cual puede ser considerado como una superclase de *Real*. Los valores del tipo *UncertainReal* son pares de números *Real* en el formato  $X = (x, u)$ , definen el valor esperado ( $x$ ) y la incertidumbre estándar asociada ( $u(x)$ ) de la cantidad  $X$ . Las operaciones sobre este tipo de datos están definidas en [2] y las hemos implementado en OCL y en Java [4].

Para calcular el coste y la duración de los procesos del metamodelo mostrado en la figura 1, se precisa contar con el coste acumulado de los recursos utilizados y consumidos en cada tarea. Estas operaciones pueden ser especificadas por las siguientes operaciones OCL: *cost()* y *duration()*. El primero suma el coste de los recursos de entrada, considerando la duración de la rama más larga, debido al paralelismo de las tareas. La operación *duration()* inicia en la tarea final del proyecto y recursivamente computa el coste máximo de todas las tareas que la han precedido.



Figura 2. Clase UML que define los atributos y operaciones del tipo UncertainReal.

```

context Process::cost() : UncertainReal
post: result = self.step->iterate(t:Step; c:Bag(UncertainReal) = Bag{} |
    c.including(t.uses.cost->sum() + t.consumes.cost->sum())->sum()

context Process::duration() : UncertainReal
post: result = self.step->select(next->isEmpty()->
    collect( t | t.accDuration()->max()

context Step::accDuration() : UncertainReal
post: result = self.duration.add(self.prev->iterate(t:Step; d:Bag(UncertainReal) =
    Bag{} | d.including(t.accDuration())->max()
    
```

Una vez obtenida la incertidumbre de los resultados, lo siguiente es calcular la incertidumbre acumulada debido a la propagación de error a través de las operaciones. La incertidumbre del coste total  $C = (c, u_c)$  del proceso es aumentado según la fórmula 1, obtenida a partir de la definición propuesta en GUM [2], la cual considera costes de los recursos utilizados ( $UR$ ) y consumidos ( $CR$ ). Esto implica que aún para los pequeños procesos, el valor mostrado en la fórmula puede representar un incremento significativo de la incertidumbre.

$$u_c = \sqrt{\sum_{i=1}^{UR} (\text{uses}_i \cdot \text{cost}.u)^2 + \sum_{j=1}^{CR} (\text{consumes}_j \cdot \text{cost}.u)^2} \quad (1)$$

Esto conlleva a estudiar la estabilidad de las medidas ante la presencia de la incertidumbre acumulada. En nuestro caso, el error relativo viene determinado por  $\eta(x) = |u(x)|/|x|$ , y lo que nos interesa es que  $\eta(x) \leq \epsilon$ , para un  $\epsilon$  pequeño y constante. Ahora bien, el análisis numérico demuestra que estas operaciones son *estables hacia atrás*; es decir, si  $X$  e  $Y$  son cantidades tales que  $\eta(x) \leq \epsilon$  y  $\eta(y) \leq \epsilon$ , y  $\otimes$  es una operación del conjunto  $\{+, -, \times, /\}$ , entonces  $\eta(x \otimes y) \leq \epsilon$ . La misma conclusión es aplicable para otras operaciones elementales tales como sqrt, exp o sin [9]. En otras palabras, el *error relativo* de los cálculos está acotado. Esto implica que el *error absoluto* crecerá con la propagación de la incertidumbre, manteniendo un nivel de crecimiento proporcional al crecimiento de las medidas agregadas. En cualquier caso es siempre preciso mantener controlado ese error absoluto pues en principio no está acotado, y eso es precisamente lo que nuestra propuesta proporciona: el cálculo de dicho error a partir de

las incertidumbre de los datos iniciales y de su propagación mediante las operaciones implementadas.

#### 4. Conclusiones y Trabajos futuros

En este artículo hemos presentado algunos de los aspectos vinculados a la representación y manejo de la incertidumbre de datos en el proceso de modelado de software. Lo interesante de nuestra propuesta es que el único cambio requerido para trabajar con la incertidumbre es el uso del tipo de dato `UncertainReal` en las variables, y el uso de las operaciones que hemos definido e implementado para modelos OCL y UML. De esta forma, la incertidumbre será considerada durante el modelado del software de una forma no intrusiva.

Existen aspectos de la incertidumbre que nos gustaría explorar en mayor profundidad, como identificar las limitaciones de nuestra propuesta. Además, introducir la incertidumbre en otros tipos de datos, y aplicarla a decisiones y selecciones múltiples. También estudiar la propuesta a otros casos y en otros dominios (por ejemplo, los sistemas ciber-físicos) para obtener mayor información sobre de la expresividad y la aplicabilidad del enfoque presentado. Finalmente, queremos refinar la metodología de uso por medio de la integración de esta propuesta en algunas herramientas de modelado y análisis (por ejemplo en Papyrus y MagicDraw).

**Agradecimientos** Este trabajo ha sido financiado por el Ministerio de Economía y Competitividad (proyecto TIN2014-52034-R "Un marco dirigido por modelos para el diseño e integración de sistemas de gestión de infraestructuras críticas") y por la Universidad de Málaga (Campus de Excelencia Internacional Andalucía Tech).

#### Referencias

1. Selic, B.: Beyond Mere Logic – A Vision of Modeling Languages for the 21st Century. In: Proc. of MODELSWARD 2015 and PECCS 2015, SciTePress (2015) IS–5
2. JCGM 100:2008: Evaluation of measurement data – Guide to the expression of uncertainty in measurement (GUM). Joint Committee for Guides in Metrology. (2008)
3. Jiménez-Ramírez, A., Weber, B., Barba, I., Valle, C.D.: Generating optimized configurable business process models in scenarios subject to uncertainty. *Information and Software Technology* (57) (2015) 571–594
4. Orue, P., Morcillo, C., Vallecillo, A.: Data uncertainty project web page (2016) [http://atenea.lcc.uma.es/index.php/Main\\_Page/Resources/DataUncertainty](http://atenea.lcc.uma.es/index.php/Main_Page/Resources/DataUncertainty).
5. JCGM 200:2012: International Vocabulary of Metrology – Basic and general concepts and associated terms (VIM), 3rd edition. Joint Committee for Guides in Metrology. (2012)
6. Mihelic, J., Mahjoub, A., Rapine, C., Robic, B.: Two-stage flexible-choice problems under uncertainty. *European Journal of Operational Research* **201** (2010) 399–403
7. Datta, P.P., Christopher, M.G.: Information sharing and coordination mechanisms for managing uncertainty in supply chains: a simulation study. *International Journal of Production Research* **49**(3) (2011) 765–803
8. Hahn, C., Friedrich, F., Winkler, T.J., Tamm, G., Petrich, K.: How to choose the right bpm tool: A maturity-centric decision framework with a case evaluation in the european market. In: EMISA. (2012) 109–122
9. Muller, J.M., et al.: *Handbook of Floating-Point Arithmetic*. Birkhäuser (2010)



# Programming Elasticity and Commitment in Dynamic Processes (Summary)

Pablo Fernandez, Hong-Linh Truong, Schahram Dustdar, and Antonio Ruiz-Cortes

University of Sevilla, Technical University of Vienna  
pablofm@us.es

**Abstract.** In the past, elasticity and commitment in business processes were under-explored. But as businesses increasingly exploit pay-per-use resources in the cloud for on-demand needs, elasticity and commitment have become important issues. Here, the authors discuss the value of using elastic resources and commitments to create more dynamic organizations that can easily balance the need to be adaptable and flexible, while also retaining a high level of manageability.  
(Published on IEEE Internet Computing, Vol 19, 2 , 68-74, 2015)

**Keywords:** elasticity, commitments, process management

## 1 Introduction

Availability, easy access to a huge number of software and human resources in the cloud (and from the crowds), and current market dynamics push organizations to scale efficiently (and save costs) but to be adaptable and meet on-demand customer needs. Such organizations search for intensive customizable solutions that could deliver specialized services while spotting and serving market opportunities through high adaptability. This leads to a clear tendency towards flexible process models that are highly reusable and adaptable, but with less manageability and control. A team-oriented, flexible process model could potentially address these challenges for adaptable organizations. However, in this organizational shift, from the process perspective two goals collide:

- the need to be adaptable and manageable, and
- the fact that adaptation is boosted by keeping a flexible process system, but manageability usually is derived from a strict process environment.

One way to deal with this dilemma is to harmonize the concepts of elasticity and commitments among individuals, teams, and organizations as the key ingredients to support efficient management of resources, which in turn leads to more adaptable process models for organizations. Elasticity allows for dynamic on-demand changes, offering functions and associated costs and quality by leveraging existing resources on site and on the cloud. Commitment represents an explicit statement of settings (that is, the resources compromised capabilities),

objectives, and the compensation model over the potential outcomes delivered. Obviously, both have a strong influence on composing and executing processes. Between them, commitment has a strong influence on elasticity, although this hasn't been well studied in dynamic processes. Here, we explore the relationships between elasticity and commitments to elaborate a list of research directions for taking elasticity into account and committing to the development of dynamic, elastic processes.

## 2 Conclusions

We need to see the importance of promoting the concepts of elastic resources and commitments, to create more dynamic organizations that can adapt their behavior while keeping a high level of manageability. Innovative management approaches that utilize elasticity and commitment could significantly improve the classical business models. Still, from a technical perspective, we must revisit concepts of process modeling, composition, and execution to be able to take into account elasticity and commitment models in Internet-scale resources.

## Servicios Web



# Monitoring the service-based system lifecycle with SALMon\*

Marc Oriol, Xavier Franch, Jordi Marco

Universitat Politècnica de Catalunya, Barcelona (Spain), GESSI research group.  
{moriol,franch}@essi.upc.edu, jmarco@cs.upc.edu

## 1 Resumen

Los Sistemas Basados en Servicios (SBS) son sistemas software altamente dinámicos compuestos por un conjunto de servicios web provenientes de distintos, y posiblemente heterogéneos, proveedores. En contraste con otros sistemas software tradicionales, el comportamiento dinámico de los SBS requiere de información actualizada sobre la calidad de servicio (QoS) para poder actuar i administrar correctamente las actividades en las distintas fases del ciclo de vida de los SBS (p.e., selección de servicios, despliegue, evaluación de niveles de acuerdo de servicio –SLA–, y adaptación).

La necesidad de proveer esta información referente a la QoS ha resultado en distintas soluciones tecnológicas construidas alrededor de un componente software de monitorización (abreviadamente, *monitor*). Sin embargo, los monitores actualmente disponibles en el estado del arte han sido diseñados para apoyar sólo un subconjunto de las actividades del ciclo de vida del SBS.

Para cubrir esta brecha de investigación, presentamos SALMon, una plataforma de monitorización de servicios versátil que provee información acerca de la QoS según la forma y enfoque adecuado para las distintas actividades del ciclo de vida.

Para ello, hemos identificado los requisitos de las distintas actividades mediante (1) el análisis de las necesidades descritas en la literatura y (2) reuniones y entrevistas con miembros de distintos grupos de investigación que trabajan en dichas actividades.

A partir de estos requisitos, hemos desarrollado SALMon. Uno de los principales obstáculos es que los requisitos de monitorización para cada tarea son distintos, y en algunos casos, pueden ser contradictorios. Esto nos ha llevado a diseñar SALMon como un sistema modular, capaz de conectar los distintos módulos de manera que permita configurar la forma final del monitor según las necesidades de cada proyecto.

Los principales avances y características que SALMon provee son los siguientes:

- Combinación de estrategias de monitorización pasiva y *testing* online.
- Combinación de estrategias de configuración basado en transformación de modelos y mediante invocación directa.
- Extensibilidad mediante el añadido de nuevos atributos de calidad.
- Bajo acoplamiento con las tecnologías de los servicios monitorizados.
- Alta interoperabilidad.

---

(\*) Artículo publicado en *Expert Systems with Applications*, 42(19), 2015. (I.F.: 2.24)

Para validar nuestra propuesta, hemos integrado SALMon con plataformas que cubren las actividades del ciclo de vida anteriormente descritas. Para ello, hemos realizado un conjunto de colaboraciones con otras universidades y centros de investigación, donde SALMon ha sido integrado con sus herramientas. En particular:

- WeSSQoS [1] (Selección de servicios): Sistema experto capaz de tratar con distintas estrategias de selección para servicios web según la QoS requerida.
- FCM [2] (Despliegue): Sistema experto para la toma de decisiones en cuanto a la mejor estrategia de despliegue en un sistema de federación de *clouds*.
- SALMonADA [3] (Validación de SLAs): Una plataforma capaz de identificar y reportar violaciones de SLAs con la descripción de las causas de violación.
- MAESoS [4], PROSA [5], PROTEUS [6], CARE [7] (Adaptación): Plataformas que dan soporte a SBS adaptativos, capaces de definir y ejecutar adaptaciones en respuesta a un mal funcionamiento o fallo de los servicios del sistema.

## 2 Agradecimientos

Este trabajo ha sido parcialmente financiado por el gobierno español mediante los proyectos CICYT TIN2010-19130-C02-01 y TIN2013-44641-P; y por la Unión Europea mediante la red de excelencia de servicios S-Cube, número de contrato 215483.

## 3 Referencias

- [1] O. Cabrera, M. Oriol, X. Franch, J. Marco, L. López, O. Fragoso, R. Santaolaya. Open framework for web service selection using multimodal and configurable techniques. In *Computación y Sistemas (CyS)*. 2014
- [2] A. Kertész, G. Keeskemeti, M. Oriol, P. Kotcauer, S. Acs, M. Rodríguez, O. Mercè, A. Cs. Marosi, J. Marco, X. Franch. Enhancing Federated Cloud Management with an Integrated Service Monitoring Approach. In *JGC*, 11(4), 2012.
- [3] C. Müller, M. Oriol, X. Franch, J. Marco, M. Resinas, A. Ruiz-Cortés, M. Rodríguez. Comprehensive Explanation of SLA Violations at Runtime. In *TSC*, 7(2), 2013.
- [4] X. Franch, P. Gruenbacher, M. Oriol, B. Burgstaller, D.Dhungana, L. López, J. Marco, J. Pimentel. Goal-driven Adaptation of Service-Based Systems from Runtime Monitoring Data, in *Procs. of COMPSACW*, 2011.
- [5] O. Sammodi, A. Metzger, X. Franch, M. Oriol, J. Marco, K. Pohl. Usage-based online testing for proactive adaptation of service-based applications, in *Procs. of COMPSAC*, 2011.
- [6] E. Schmieders, A. Micsik, M. Oriol, K. Mahhub, R. Kazhamiak. Combining SLA prediction and cross layer adaptation for preventing SLA violations, in *Procs. of 2<sup>nd</sup> WoSS*, 2012.
- [7] M. Oriol, N. Qureshi, X. Franch, A. Perini, J. Marco. Requirements monitoring for adaptive service-based applications, in *Procs. of REFSQ*, 2012.

# Abstract of “Modelling Service Level Agreements for Business Process Outsourcing Services”\*

Adela del-Río-Ortega, Antonio Manuel Gutiérrez, Amador Durán, Manuel Resinas, and Antonio Ruiz-Cortés

Universidad de Sevilla, Spain

*Service level agreements* (SLAs) have been used by many proposals in the last decade to automate different stages of the service lifecycle, using a formal definition of the different parts of an SLA to automate their negotiation [1], the provisioning and enforcement of SLA-based services [2], or the monitoring, explanation and prediction of SLA runtime violations [3, 4]. Most of them have been designed for computational services. Therefore, they are aimed at enhancing software that supports the execution of computational services such as network monitors or application servers with SLA-aware capabilities, where there are no human or non-automatic tasks involved in service consumption.

On the other hand, *business process outsourcing* (BPO) services are non-computational services such as logistics, supply-chain, or IT delivery services, that are based on the provisioning of business processes as services, providing partial or full business process outsourcing. Like computational services, their execution is regulated by SLAs and supported by specific software [5]. In this case, since BPO services are process-oriented, the software that supports them is usually a *process-aware information systems* (PAIS) such as ERPs, CRMs, or BPMs. However, unlike computational services, there is little work related to the extension of PAIS with SLA-aware capabilities to support BPO services.

An SLA-aware PAIS is a PAIS that uses explicit definitions of SLAs to enable or improve the automation of certain tasks related to both the SLAs and their fulfilment, such as performance monitoring, human resource assignment or process configuration [6]. For instance, an SLA-aware PAIS could be automatically instrumented according to the metrics defined in the SLA so that when there is a risk of not meeting a *service level objective* (SLO), an alert is raised allowing the human actors involved in the process to take measures to mitigate the risk. Another example could be the automated configuration of the process, *e.g.* removing or adding activities, executed by the SLA-aware PAIS depending on the conditions of the SLA agreed with the client.

Apart from the benefits derived from the automation of these tasks, the need for an SLA-aware PAIS becomes more critical in a BPO scenario, where the

\* A. del Río-Ortega et al.: Modelling Service Level Agreements for Business Process Outsourcing Services. In: CAiSE 2015: 485-500. This work has received funding from the European Commission (FEDER), the Spanish and the Andalusian R&D&I programs (grants TIN2012-32273 (TAPAS), TIC-5906 (THEOS) and P12-TIC-1867(COPAS)).

conditions of the SLA agreed with each client may vary, becoming it crucial for the PAIS that supports it to behave according to the SLA agreed with the client. An example could be the prioritisation of the execution of tasks for those clients whose SLAs have bigger penalties if they are not met.

In this paper, we focus on the formalization of BPO SLAs as a first step to enable such SLA-aware PAIS. After a thorough analysis of the SLAs related literature for computational and non-computational services, and of more than 20 BPO SLAs from different organisations, four main aspects involved in their formalization have been identified: 1) the description of the business process provided by the service; 2) the SLOs guaranteed by the SLA; 3) the penalties and rewards that apply if guarantees are not fulfilled; and 4) the definition of the metrics used in these guarantees. According to this analysis, the general structure of SLAs for BPO services and the definition of SLOs, penalties, and rewards are very similar to those of SLAs defined for computational services. However, the service description and the definition of the SLA metrics of BPO SLAs and computational SLAs present significant differences. The reason is that, unlike computational services, BPO services are process-aware and this has a strong influence on how they are described.

On the light of these requirements, our proposal to model BPO SLAs combines well founded approaches and standards for modelling computational SLAs and process performance indicators. Specifically, we rely on WS-Agreement to provide the general SLA structure, BPMN to model the business process related to the service, PPINOT [7] for the definition of metrics, and iAgree [8] to define SLOs and penalties.

The application of the proposed approach to a number of real scenarios allowed us to conclude that our approach is able to model all possible situations in these scenarios except for some limitations concerning the definition of SLA metrics, that shall be carried out in future work.

## References

1. M. Resinas et al.: A bargaining-specific architecture for supporting automated service agreement negotiation systems. *Sci Comput Program* **77**(1) (2012) 4–28
2. V. C. Emeakaroha et al.: Cloud resource provisioning and SLA enforcement via LoM2HiS framework. *Concurr Comp-Pract E* **25**(10) (2013) 1462–1481
3. C. Müller et al.: Comprehensive Explanation of SLA Violations at Runtime. *IEEE Transactions on Services Computing* **7**(2) (2014) 163–183
4. P. Leitner et al.: Data-driven and automated prediction of sla violations in service compositions. *Distributed and Parallel Databases* **31**(3) (2013) 447–470
5. A. M. Gutiérrez et al.: Extending ws-agreement to support automated conformity check on transport and logistics service agreements. In: *ICSOC*. (2013) 567–574
6. W. M. P. van der Aalst: Business process configuration in the cloud: How to support and analyze multi-tenant processes? In: *ECOWS*. (2011) 3–10
7. A. del-Río-Ortega et al.: On the Definition and Design-time Analysis of Process Performance Indicators. *Information Systems* **38**(4) (2012) 470–490
8. C. Müller et al.: Automated Analysis of Conflicts in WS-Agreement. *IEEE Transactions on Services Computing* (2013) 1–1



# Supporting Compensations with WS-Agreement\*

Carlos Müller, Pablo Fernandez, Octavio Martín-Díaz, Antonio M. Gutierrez,  
Manuel Resinas, and Antonio Ruiz-Cortés

University of Sevilla

**Abstract.** During the last years the use of service level agreements (SLA) is rising uncontrollably to describe the rights and obligations of parties involved in service provisioning (typically the service consumer and the service provider); amongst other information, SLA could define guarantees associated with the idea of service level objectives (SLOs) that normally represent key performance indicators of either the consumer or the provider. In case the guarantee is under or over fulfilled SLAs could also define some compensations (i.e. penalties or rewards). In such a context, there have been important steps towards the automation of the analysis of SLAs. One of these steps is a characterization model of SLAs with compensations proposed by the authors in a previous work; and another step is the standardisation effort in the SLAs notation made by WS-Agreement. However, real-world SLAs includes complex concepts that must be considered, namely: (i) SLA terms that specify compensations without an explicit SLO; and (ii) a limit for the compensations. In this paper we extend our prior characterization model considering these complex concepts. Specifically, (i) we provide up to five real-world scenarios whose SLAs incorporate aforementioned new concepts; (ii) we extend our model for compensable guarantees considering terms without an explicit SLO; and (iii) we provide a novel WS-Agreement-based syntax to model SLAs with compensations considering these concepts. These contributions aim to establish a foundation to elaborate tools that could provide an automated support to the modelling and analysis of SLAs with compensations.

## 1 Introduction

In the recent years the use of service level agreements (SLAs) is in continuous rising to describe the parties rights and obligations to support a reliable service consumption. Specifically, SLAs are composed by different terms that typically define guarantees associated with a certain service level objective (SLOs) and they should be enforced by one party (the guarantor) to another party (the beneficiary); in most cases the former correspond to the service provider, and the latter to the service consumer. For instance, Amazon as provider of their Web

---

\* This work was partially supported by the Spanish and the Andalusian R&D&I programmes (grants P12-TIC-1867, TIN2015-70560-R).

Services (AWS), guarantees an availability  $\geq 99.95\%$  as general SLO. Additionally, real-world SLAs usually include a set of compensations that represent the consequences of underfulfilling (penalties) or overfulfilling (rewards) the SLOs. For instance, Amazon is penalised if the availability drops below 99.95%. In a previous work [3] we coined the concept of Compensable SLAs referring to such SLAs that include at least a compensation action, either a penalty or a reward.

In such a context, there have been important steps towards the automation of the analysis of SLAs. One of these steps is a characterization model of SLAs with compensations proposed in [3]; and another step is the standardisation effort of WS-Agreement to provide a notation for the SLAs. However, real-world SLAs are thoroughly described and therefore they include many complex concepts that must be considered in the SLA models and notations, namely: (i) guarantees defined by several regional government of Canada and Spain specify compensations without an explicit SLO; and (ii) a limit for the compensations that companies as Amazon establish to compensate consumers of the Elastic Cloud service (AWS EC2) up to a limit of the 10% of the EC2 monthly bill<sup>1</sup>.

In this paper we perform a decisive step towards automating the analysis of Compensable SLAs by modelling them considering the aforementioned complex concepts as follows. First, we incorporate to our previous characterization model the definition of *Optimal Thresholds* that can be inferred depending on whether the SLA was specified by the guarantor or the beneficiary. These thresholds are used to validate compensations when the guarantees do not include an explicit SLO. And second, we propose a novel and user friendly notation to specify compensations within the *iAgree syntax*, a WS-Agreement-based [1] language [8, 9]. The advantage of extending iAgree is to leverage its existing analysis tooling support that already incorporates the notion of consistence and validity of SLAs without compensations [8].

This paper is structured as follows: section 2 summarises the characterisation model provided for Compensations in [3]. Such a characterisation model is extended with the Optimal Threshold definition in section 3. In section 4 we propose our iAgree syntax extension to model compensations. Our modelling proposal is validated in five real-world scenarios in section 5. In section 6 we analyse the literature to identify related approaches dealing with compensations. Finally, in section 7 we outline some conclusions and future work.

## 2 Compensations Model in a Nutshell

In this section we summarised the compensations model we proposed in [3] comprising: (i) the conceptualization of the Compensation Function to express consistently penalties and rewards and (ii) a model for Compensable Guarantees that associate SLOs with Compensation Functions.

---

<sup>1</sup> <http://aws.amazon.com/es/ec2/sla/>

## Compensation Functions

Compensation Functions (CF) are defined over services properties in the context of a guarantee satisfied by a guarantor to a beneficiary. Specifically, they associate two types of compensations depending on the subject and recipient of the compensations: on the one hand, a penalty represents a compensation from the guarantor to the beneficiary and, on the other hand a reward represents a compensation from the beneficiary to the guarantor. Following we formalize CFs by means of a set of supporting core definitions:

**Definition 1 (Service Property Values).** *The set  $SP_{sp}$  denotes the set of all possible values of a service property  $sp$  ( $SP_{sp} = \{v_1, \dots, v_n\}$ ).*

**Definition 2 (Utility Function).** *An Utility Function for a certain service property  $sp$ , denoted by  $UF_{sp}$ , is a function from  $SP$  to  $\mathbb{R}$  that associates a utility to each of the values; i.e. it defines which service properties values  $SP_{sp}$  are more interesting for a given party.*

**Definition 3 (Utility Precedence).** *Let  $v_1$  and  $v_2$  be values of the set  $SP_{sp}$  of a service property  $sp$ , and  $UF_{sp}$  a utility function defined on the same service property; a precedence relation called utility precedence is defined on  $SP_{sp}$  by  $UF_{sp}$ . Thus, we denote that  $v_1$  is less interesting than  $v_2$  by  $v_1 \prec v_2$ .*

**Definition 4 (Compensation Function).** *A compensation function for a given service property  $sp$ , denoted by  $CF_{sp}$ , is a function from  $SP$  to  $\mathbb{R}$  that associates a compensation to each of the values.*

**Definition 5 (Compensation Regions).** *A compensation function for a given service property  $sp$   $CF_{sp}$  defines up to three compensation regions:  $Penalized(CF_{sp}) = \{v_i \in SP_{sp} \cdot CF_{sp}(v_i) > 0\}$ ,  $Neutral(CF_{sp}) = \{v_i \in SP_{sp} \cdot CF_{sp}(v_i) = 0\}$ , and  $Rewarded(CF_{sp}) = \{v_i \in SP_{sp} \cdot CF_{sp}(v_i) < 0\}$ .*

Figure 1 shows a typical compensation function with the regions.

**The Validity of CFs** is formalized as a property related with the consistence in terms of utility and the saturability of compensations.

*Property 1 (Consistent<sub>CF</sub>).* *A compensation function  $CF_{sp}$  is said to be consistent if the compensation for a less interesting value of service property is less or equal than the compensation for a more interesting value according with the utility precedence defined by the Utility function of the beneficiary.*

$Consistent_{CF}(CF_{sp}) \iff \forall v_1, v_2 \in SP \cdot v_1 \preceq v_2 \Rightarrow CF_{sp}(v_1) \geq CF_{sp}(v_2)$

*Property 2 (Saturated).* *A compensation function  $CF_{sp}$  is said to be saturated if there exist two values ( $v_{min}$  and  $v_{max}$ ) for the service property, that delimit the higher compensation, either penalty or reward.  $Saturated(CF_{sp}) \iff \forall v_i \in SP, \exists v_{max}, v_{min} \in SP \cdot CF_{sp}(v_i) \leq CF_{sp}(v_{max}) \wedge CF_{sp}(v_i) \geq CF_{sp}(v_{min})$*

*Property 3 (Valid<sub>CF</sub>).* *A compensation function  $CF_{sp}$  is said to be valid if it is consistent and saturated.  $Valid_{CF}(CF_{sp}) \iff Consistent_{CF}(CF_{sp}) \wedge Saturated(CF_{sp})$*

### Compensable SLA

A key element of SLAs are the guarantee terms [1] that are typically defined over a service level objective. Based on this conceptualization, we coin the concept of Compensable Guarantees (CG) to those which include a CF and subsequently, Compensable SLAs represent a type of SLA that includes at least one CG.

**Definition 6 (Service Level Objective).** An  $SLO_{sp}$  is a valid<sup>2</sup> assertion defined over a service property  $sp$ .

**Definition 7 (Fulfillment Regions).** The assertion defined by an  $SLO_{sp}$  determines two regions over the values of the service properties they dealt with:  $Fulfilled(SLO_{sp}) = \{v_i \in SP_{sp} \cdot SLO_{sp}\}$ , and  $Unfulfilled(SLO_{sp}) = \{v_i \in SP_{sp} \cdot \neg SLO_{sp}\}$ . This regions are delimited by the threshold  $Th_{SLO}$ .

**Definition 8 (Compensable Guarantee).** A compensable guarantee  $CG_{sp}$  is a two-tuple of the form  $(SLO_{sp}, CF_{sp})$  in which  $SLO_{sp}$  is a service level objective and  $CF_{sp}$  is a compensation function that are defined over the same service property  $sp$ .  $CG_{sp} = \langle CF_{sp}, SLO_{sp} \rangle$

Figure 1 depicts the relationships between the fulfillment regions delimited by the SLO (cf.  $Th_{SLO}$ ) and the compensation regions defined by the CF.

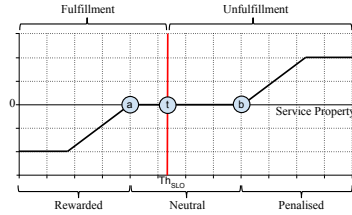


Fig. 1. Generic CF with CG showing compensation and fulfillment regions.

**The Validity of CGs** is formalized as a property related with the consistency between its SLO and its CF.

*Property 4 (Consistent).* A compensable Guarantee  $CG_{sp}$  is said to be consistent if there is at least one fulfilled and neutral value (that would be  $Th_{SLO}$ ) and the fulfillment regions are coherent with compensation regions: the fulfilled values are either neutral or rewarded and, complementary, the unfulfilled values are either neutral or penalized.

*Property 5 (Valid).* Let  $CG_{sp}$  a compensable guarantee, it is said to be valid if it is consistent and it contains a valid CF.

$$Valid_{CG}(CG_{sp}) \iff Valid_{CF}(CG_{sp}.CF) \wedge Consistent_{CG}(CG_{sp})$$

<sup>2</sup> A formal validity criteria for SLOs is presented in [8].

### 3 Extending the Compensable Guarantees Model

Some compensable guarantees of public SLAs do not explicitly express a particular SLO and the guarantee is only expressed in terms of compensations (c.f. GNWT and IT\_AHS scenarios in the following section 5). In these cases two optimal<sup>3</sup> thresholds can be inferred to delimit the fulfillment regions.

**Definition 9 (Optimal Thresholds).** *Let  $CG_{sp}$  a compensable guarantee without defining an explicit SLO  $SLO_{sp}$ , two optimal thresholds can be inferred: (i) the optimal threshold for the guarantor ( $Th_{Gtor}$ ) that corresponds with the less compensable service property value for the beneficiary that does not involve a penalty (e.g. value  $b$  in Figure 1) and (ii) the optimal threshold for the beneficiary ( $Th_{Ben}$ ) that corresponds with the more compensable service property value for the beneficiary that does not involve a reward (e.g. value  $a$  in Figure 1).*

$$Th_{Gtor} = v_j | \forall v_i, v_j, v_k \in SP. \quad v_i \preceq v_j \prec v_k \wedge \\ CF_{sp}(v_i) \leq 0 \wedge CF_{sp}(v_j) = 0 \wedge CF_{sp}(v_k) > 0$$

$$Th_{Ben} = v_j | \forall v_i, v_j, v_k \in SP. \quad v_i \prec v_j \preceq v_k \wedge \\ CF_{sp}(v_i) < 0 \wedge CF_{sp}(v_j) = 0 \wedge CF_{sp}(v_k) \geq 0$$

These optimal thresholds are required to check the compensable guarantee validity when they do not define explicitly the SLO. Moreover, the optimal thresholds are also useful for checking if an SLO of a compensable guarantee has been optimally defined in term of compensations. i.e., an SLO is optimally defined if the SLO and the optimal threshold delimit the same fulfillment regions.

### 4 Extending iAgree with Compensations

The main reason for extending iAgree to model compensations is the possibility of updating its tooling support. Currently, iAgree is supported by IDEAS<sup>4</sup>, a service based application with provides, between others, analysis and edition facilities.

Figure 2 includes the proposed iAgree syntax for specifying compensations. Specifically, it comprises: (i) the interval (e.g. `monthly`, `weekly`, etc) in which a compensation is applied; (ii) an optional compensation limit that is an expression to cap the amount to compensate; and (iii) the compensation itself, that is applied when a compensation condition is fulfilled.

<sup>3</sup> Note that these thresholds are optimal in terms of compensations, but not necessary in terms of cost or any other SLO-related aspect.

<sup>4</sup> Available at <http://www.isa.us.es/IDEAS/>

```

Agreement ident version ...
AgreementTerms ...
  Guarantee Terms
    guaranteeId: Provider/Consumer guarantees SLO; //id , obligated , SLO
    with interval penalty/reward //interval , compensation type
    [upTo compensationLimit;] //compensation limit
    of compensation if compCondition; //compensation , condition
    ...;
  end

```

Fig. 2. Schema of compensable guarantee terms in *iAgree* syntax

## 5 Modelling Compensable SLAs of Real-World Scenarios

In this section we introduce each scenario, we present an example of compensation identified in its SLA, and we model it with our proposed *iAgree* extension. Due to the space limit, we do not include all the presented examples in *iAgree*, but all of them are publicly available in an IDEAS demo workspace<sup>5</sup>. In addition, some scenarios (c.f. AWS EC2 and GNWT) also include a figure showing the guarantee term and the corresponding CF (as a black line) along with the  $Th_{SLO}$  (as a solid vertical line) derived from the SLO; in case there is no explicit SLO, we have inferred an optimal threshold ( $Th_{Gtor}$  or  $Th_{Ben}$ ) depending on whether the SLA was specified by the guarantor (AWS EC2 scenario) or the beneficiary (GNWT scenario); these optimal thresholds are depicted as discontinuous lines. While dark points in the figures denote the inclusion of the service property value in the interval, gray points mean the value exclusion.

**AWS EC2 SLA.** Amazon Web Services (AWS) is a service catalogue that has boosted the idea of cloud computing in the industry; amongst them, the Elastic Computing Cloud (EC2) represents a widely used Infrastructure as a Service.

Amazon has explicitly published an SLA for EC2<sup>6</sup> that is based on the idea of Monthly Uptime Percentage (MUP); this element characterizes a guarantee over the availability of the virtual resources requested. Specifically, the consequences of failing a certain MUP is defined by Amazon in two levels: in case the MUP drops below 99.95% and in case the MUP drops below 99%. Figure 3(a) depicts the CF of this scenario that is defined as a percent of discount in the next billing cycle a.k.a Service Credit Percentage (SCP). The figure shows that the SLO threshold does not coincide with the optimal threshold of the guarantor in term of compensations. As denoted in the *iAgree* document of Figure 3(b), Amazon compensates consumers up to a limit of the 10% of the EC2 monthly bill.

<sup>5</sup> <https://labs.isa.us.es:8181/IDEAS/demo/Compensations>

<sup>6</sup> Available at <http://aws.amazon.com/es/ec2/sla/>

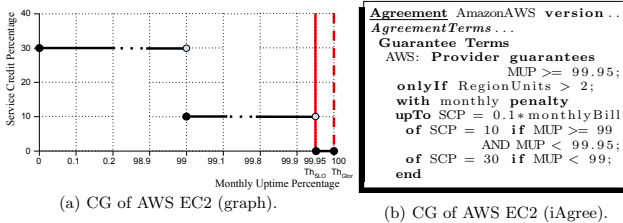


Fig. 3. Compensable Guarantee of AWS EC2 in graph and iAgree.

**Telecomm SLA.** The regional Government of Andalusia in Spain outsources the installation and management of telecommunication networks. An SLA<sup>7</sup> is specified by the government including some penalties for the services provider.

The term (ARG-1) selected from the agreement specifies an SLO demanding that 90% of interventions must be solved. However, some penalties apply for a range of values that fulfill such a demand. Specifically, if the service provider solve more than 95% of interventions no penalties apply, but some bill penalties apply from 90% to 95% of interventions solved. This situation implies a definition error [3]. In addition, compensations are capped at the 20% of the monthly bill.

**GNWT SLA.** The Government of the Northwest Territories (GNWT) of Canada outsources the IT support. They provide an SLA<sup>8</sup> with the desired guarantees and compensations. Two examples of terms whose CFs are included in Figure 4 have been extracted from its SLA, and Figure 5 specifies it using iAgree syntax. Example GNWT-2 depicts specific times for different milestones that take place in the resolution of problems that have made a critical application function unusable or unavailable and no workaround exists (severity 1 code). In this case, a reward for the provider applies if all problems are resolved in less than 2 hours, and a penalty for the provider applies if any of them is resolved in more than 4 hours. This SLA also includes a term relating the scheduled project delivery and the real project delivery that is shown in example GNWT-4. This term includes a reward for the provider if the elapsed days until delivery are less than 20% lesser than planned but also a penalty for the provider if the elapsed days until delivery are exactly 20% greater than planned. In this case the penalty is not correctly defined [3]. Although an SLO is not explicit in this case, the iAgree document of Figure 5 includes the inferred optimal threshold as guarantee.

<sup>7</sup> Available at <http://goo.gl/Wike8y>

<sup>8</sup> Available at <http://www.fin.gov.nt.ca/ocio/sim/sdlc/3/resources/sla.htm>

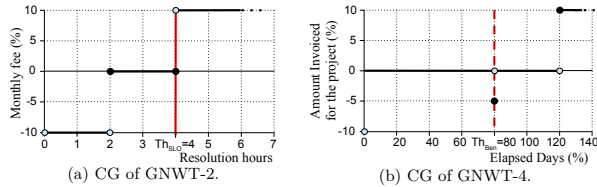


Fig. 4. Compensable Guarantees of GNWT examples.

```

Agreement GNWT version ...
AgreementTerms ...
  Guarantee Terms ...
    GNWT-2: Provider guarantees resolutionHours <= 4;
      onlyIf SeverityType = 1;
      with monthly reward
        of feeRewardPercentage = 10 if resolutionHours <= 2;
      end
      with monthly penalty
        of feePenaltyPercentage = 10 if resolutionHours > 4;
      end
    ...
    GNWT-4: Provider guarantees elapsedDaysP < 120;
      with monthly reward
        of invoiceRewardP = 5 if elapsedDaysP = 80;
      end
      with monthly penalty
        of invoicePenaltyP = 10 if elapsedDaysP >= 120;
      end
  
```

Fig. 5. Compensable guarantee terms of GNWT scenario in *iAgree* syntax

**IT\_AHS.** The regional Government of Andalusia in Spain outsources the IT support services for the public health service. An SLA<sup>9</sup> is specified by the regional government including guarantees on the services and defining the applicable penalties. For instance, it is stated that if the availability of the continuous IT improvement service is less than 95%, it is defined a penalty that depends on the hourly price of the demanded human resource profile. In addition, the SLA caps penalties either at 25% of monthly charge, or at 10% of total budget.

**Verizon Terremark Cloud Backup Service.** Verizon Terremark offers a variety of cloud services being one of them a cloud backup service<sup>10</sup>. An SLA<sup>11</sup> is specified to assure an availability greater than 99.9% under the penalty of a service credit (defined as 1/30 of the cloud backup services monthly fee) plus one

<sup>9</sup> Available at <http://ow.ly/4mO6cB>

<sup>10</sup> <http://ow.ly/4mORhQ>

<sup>11</sup> Available at <http://ow.ly/4mOFFd>



credit more for each 100 extra minutes in which the availability is not restored. However, such a penalty cannot exceed the 50% of the monthly fee.

## 6 Related Work

As far as we know, there is no proposal to model compensation in SLAs which enables automating its analysis.

The proposal of Leitner et al. in [6] formalizes the problem of finding the optimal set of adaptations, which minimizes the total costs arising from SLA violations and the adaptations to prevent them. In this work, a model for penalty functions is presented; this formalization has been the starting point of our characterisation model for compensations presented in [3] and summarised in Section 2. In [10] the same authors present an approach for optimally scheduling incoming requests to virtual computing resources in the cloud, so that the sum of payments for resources and loss incurred by SLA violations is minimized. The example relates the penalty with a service property representing the duration of requests to virtual computing resources in the cloud.

Other examples are the following: Buco et al. propose in [7] an SLA management system, called SAM that provides penalties in a Service Level Management process. Grabarnik et al. propose in [4] a model that can be used to reduce total service costs of IT service providers using alternative delivery teams and external service providers. Rana et al. identifies in [12] how SLOs may be impacted by the choice of specific penalty clauses. Paschke et al. [11] model an SLA to automate its management. This SLA defines minimum and maximum thresholds to compensate SLAs underfilling or overfilling, but this compensation is ad-hoc modelled through event calls.

In business studies, utility function models are also analysed as they are strongly dependent on customer preferences and behaviour. [2] describes a business scenario with cost, customer expectations and reputation variables where reward function follows a non-monotonic behaviour (based on satisfying preferences from different customers). Similarly, Fenghui Ren et al. analyse in [13] how utility function is obtained from customer objective function (i.e., customers timetable preferences affect how transactions distribute through commercial opening hours).

Angelov et al. propose in [5] a formal representation for contracts to detect and solve different kinds of conflicts. Although the proposed contracts representation supports penalties and rewards by means of reparation clauses, they are not validated against utility functions as proposed in the current paper.

## 7 Conclusions and Future Work

In this paper we contribute with a way of modelling compensable SLAs by extending iAgree, a WS-Agreement-based [1] language [8, 9]. The extended iAgree notation supports: (i) a previously proposed formal model of compensations [3]; (ii) the specification of compensations limits that is a common practice as shown

in the five real-world scenarios included in the paper. In addition, we have also considered those cases in which the SLOs are not explicit in the SLA guarantees. Thus, we have defined the optimal thresholds that can be inferred depending on whether the SLA was specified by the guarantor or the beneficiary.

The contribution constitutes an important step towards the automation of analysis of Compensable SLAs because iAgree has a sounded analysis tooling support framework called IDEAS that is publicly available. Moreover, a public IDEAS demo workspace including all compensable SLA examples of the paper modelled with iAgree has been prepared. As future work we will update the analysis operations of IDEAS to validate both, compensation functions and compensable guarantees, in order to detect singular situations and common pitfalls. Such an update will consist on using mathematics libraries and the already used constraints satisfaction problems solvers to check the properties of our compensation model.

## References

1. Andrieux et al. Web Services Agreement Specification (WS-Agreement) (v. gfd-r.192), 2011. OGF - Grid Resource Allocation Agreement Protocol WG.
2. Heski Bar-Isaac and Joyee Deb. What is a good reputation? career concerns with heterogeneous audiences. *Journal of Industrial Organization*, 34(0):44 – 50, 2014.
3. Carlos Müller et al. *Towards a Formal Specification of SLAs with Compensations*. *OTM Conferences: CoopIS, Amantea, Italy*, pages 295–312. 2014.
4. G. Grabarnik et al. Management of service process qos in a service provider - service supplier environment. In *IEEE Int. Conf. on Ent. Comp., E-Commerce, and E-Services. (CEC/EEE)*., pages 543–550, July 2007.
5. Krasimir Angelov et al. A framework for conflict analysis of normative texts written in controlled natural language. *The Journal of Logic and Algebraic Programming*, 82(57):216 – 240, 2013.
6. P. Leitner, W. Hummer, and S. Dustdar. Cost-based optimization of service compositions. *Services Computing, IEEE Transactions on*, 6(2):239–251, April 2013.
7. M.J. Buco et al. Utility computing sla management based upon business objectives. *IBM Systems Journal*, 43(1):159–178, 2004.
8. C. Müller, M. Resinas, and A. Ruiz-Cortés. Automated analysis of conflicts in ws-agreement. *IEEE Transactions on Services Computing (TSC)*, 2013.
9. Carlos Müller. *On the Automated Analysis of WS-Agreement Documents. Applications to the Processes of Creating and Monitoring Agreements*. International dissertation, Universidad de Sevilla, 2013.
10. P. Leitner et al. Cost-Efficient and Application SLA-Aware Client Side Request Scheduling in an Infrastructure-as-a-Service Cloud. In *Cloud Computing (CLOUD)*, *IEEE 5th International Conference on*, pages 213–220, June 2012.
11. Adrian Paschke and Martin Bichler. Knowledge representation concepts for automated SLA management. 23 November 2006.
12. O. F. Rana, M. Warnier, T. B. Quillinan, F. Brazier, and D. Cojocararu. Managing violations in service level agreements. In *Grid Middleware and Services Chapter Title - Managing Violations in Service Level Agreements*, pages 349–358, 2008.
13. Fenghui Ren and Minjie Zhang. Bilateral single-issue negotiation model considering nonlinear utility and time constraint. *Decision Support Systems*, 60(0). 2014.

# MigraSOA: Migrando aplicaciones web legadas hacia arquitecturas orientadas a servicios (SOA)\*

Encarna Sosa-Sánchez, Pedro J. Clemente, Álvaro Prieto, José M. Conejero,  
and Roberto Rodríguez-Echeverría

Universidad de Extremadura. Quercus Software Engineering Group,  
{esosa, pjcclemente, aeprieto, chemacm, rre}@unex.es,  
<http://quercusseg.unex.es>

**Resumen** La migración de aplicaciones legadas hacia arquitecturas orientadas a servicios (SOA) es un proceso relativamente habitual en la actualidad, ya que las características de flexibilidad arquitectónica que ofrece SOA permiten adaptar fácilmente las aplicaciones a los nuevos requisitos marcados por las empresas. Sin embargo, el desarrollo de esta migración hacia estas nuevas arquitecturas software se lleva a cabo normalmente de forma manual, siendo este un mecanismo tedioso y propenso a errores. MigraSOA es una propuesta de migración de aplicaciones web legadas (LWA) hacia SOA que utiliza técnicas de Desarrollo de Software Dirigido por Modelos (MDD) para abordar la complejidad de las tecnologías subyacentes (servicios web, definición de procesos de negocio o plataformas para procesos de negocio ejecutables). En este trabajo, además de presentar MigraSOA de una forma global, nos centraremos en los aspectos de alineación de los procesos de negocio definidos por la empresa con los servicios web subyacentes en la aplicación legada y en cómo extender los modelos BPMN para conseguir la sincronización entre ellos y los servicios disponibles.

**Keywords:** migración de aplicaciones web, servicios web, procesos de negocio, arquitecturas orientadas a servicios

## 1 Introducción

Las empresas en general están sometidas a constantes cambios en sus procesos de negocio, lo cual requiere un esfuerzo importante para mantener sincronizados los procesos de negocio y su infraestructura tecnológica [14]. No obstante, las empresas deben rentabilizar las inversiones que en su momento realizaron en esta infraestructura tecnológica y además implementar la funcionalidad necesaria para soportar dichos cambios en sus procesos de negocio. En muchos casos

---

\* This work was funded by Ministerio de Economía y Competitividad (Spain) - Project TIN2015-6957-R; the Junta de Extremadura, Consejería de Economía e Infraestructuras under grant GR15098 and by the European Regional Development Fund (ERDF).

esta infraestructura tecnológica está soportada por aplicaciones web que hoy en día pueden considerarse obsoletas y que deben ser rediseñadas para ofrecer arquitecturas más flexibles y que puedan responder a este continuo cambio de requisitos.

Además, las compañías deben definir sus procesos de negocio que posteriormente serán implementados mediante servicios ad hoc o adquiridos de terceros, convirtiendo el desarrollo de software en un auténtico proceso de composición de servicios interoperables [6]. En este contexto, las arquitecturas orientadas a servicios (SOA) han supuesto una auténtica revolución para el desarrollo de sistemas complejos, que integren servicios desarrollados bajo múltiples plataformas y que deben coordinarse para satisfacer los procesos de negocio de las compañías [6, 11].

Llegados a este punto, por una parte encontramos aplicaciones web legadas susceptibles de ser modernizadas y, por otra, una filosofía de desarrollo de software, SOA, que facilita la construcción de aplicaciones interoperables, extensibles, fiables, disponibles, escalables y adaptables [8].

La modernización de aplicaciones legadas ha tendido hacia la exposición de sus funcionalidades como servicios web [2, 3, 5, 7, 10, 17], sin embargo, en la mayoría de estas modernizaciones no se aborda la sincronización de los procesos de negocio de las empresas con estos servicios. En [9] se aboga por una propuesta dirigida por modelos para realizar la migración de aplicaciones web legadas hacia *Rich Internet Applications* (RIAs). Este trabajo ofrece un mecanismo de ingeniería inversa para abordar el análisis del código de la aplicación web legada (LWA) y obtener un modelo MVC (Modelo Vista Controlador) que la representa conceptualmente. Ello habilita la utilización de técnicas de desarrollo dirigido por modelos para modernizar la LWA hacia una nueva arquitectura, ya que a partir de la información disponible en dicho modelo, podremos abordar las transformaciones modelo a modelo y modelo a código necesarias para avanzar en la nueva versión de la LWA, alineada con los procesos de negocio de la compañía.

Así, a partir de [9], en [12, 13] se han presentado los aspectos esenciales de MigraSOA, un marco para la modernización de aplicaciones web legadas hacia SOA. La propuesta MigraSOA además de ofrecer un mecanismo dirigido por modelos para representar conceptualmente la aplicación orientada a servicios y generar los servicios web que actuarán como envoltorios (*wrappers*) sobre la LWA, también aborda la alineación de procesos de negocio (definidos por las compañías con los servicios web legados (obtenidos a partir las aplicaciones web migradas). Las principales contribuciones de este trabajo son las siguientes:

- Definición del marco de modernización de aplicaciones web legadas denominado MigraSOA.
- Alineación de los procesos de negocio definidos en BPMN con los servicios web subyacentes que encapsulan la funcionalidad de la LWA.
- Extensión de los procesos de negocio definidos en BPMN e incorporación automática del código necesario para convertirlos en BPMN ejecutables, los cuales podrán interactuar con los servicios web obtenidos en el proceso MigraSOA.

La estructura del artículo se resume a continuación. En la sección 2 se presenta de forma general la propuesta MigraSOA. En la sección 3 se describe la definición de procesos de negocio y el proceso de alineación de los mismos con los servicios web de la LWA. Posteriormente, en la sección 4 se describe el proceso de extensión de los modelos BPMN para convertirlos en BPMN ejecutables. En la sección 5 se presentan las conclusiones obtenidas y los trabajos futuros.

## 2 MigraSOA: Migración semi-automática hacia SOA

Con el objetivo de hacer que este trabajo pueda ser presentado en su contexto, en esta sección se presenta una breve descripción de las cuatro fases principales que han sido definidas en la propuesta de migración LWA hacia SOA [12]. Estos pasos pueden observarse en la Figura 1 y se resumen en:

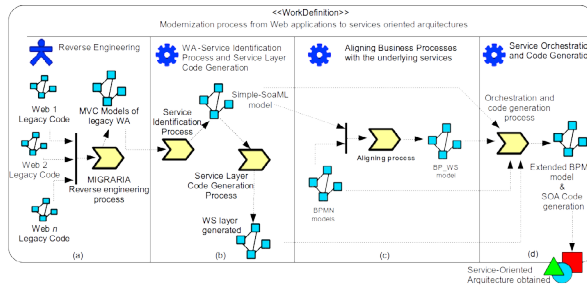


Fig. 1. Esquema de la migración de aplicaciones web legadas hacia SOA

- a) Ingeniería inversa. El punto de partida de esta propuesta es una fase (identificada como *a* en la Figura 1) en la que, mediante el análisis estático de los artefactos software de una LWA, implementado con técnicas y herramientas de reingeniería, se obtiene su representación conceptual conforme al metamodelo MIGRARIA-MVC, desarrollado en el proyecto MIGRARIA [9].
- b) Identificación de servicios y generación de código de la capa de servicios. En esta fase (*b* en la Figura 1) se refina el modelo conceptual etiquetando los servicios localizados en base a una búsqueda de determinados patrones (ejemplo de servicios podrían ser: buscar un artículo por su ID, añadir un artículo, comprobar el estado de un artículo, etc.). Este paso es importante para el proyecto completo, ya que una apropiada identificación de los servicios permite generar adecuadamente la capa de servicios. El modelo

obtenido en este paso (Simple-SoaML model) es conforme con un metamodelo denominado Simple-SoaML [12]. A partir de este modelo, se ha definido una transformación modelo a texto que permite generar de forma automática el código correspondiente a la capa de servicios subyacentes, la cual facilita, mediante servicios web tipo SOAP, interactuar con la funcionalidad de la LWA [12].

- c) Alineación de procesos de negocio con los servicios subyacentes. El objetivo de esta fase (*c* en la Figura 1) es alinear los procesos de negocio definidos por la compañía con la capa de servicios generada en la fase anterior, para ello, se implementa un proceso de alineación que toma como entrada, por una parte, los modelos de procesos de negocio de la empresa, definidos en BPMN y, por otro lado, el modelo Simple-SoaML (obtenido en la fase *b*). El resultado de este proceso de alineación es un modelo denominado BP-WS que relaciona las tareas de los procesos de negocio con los servicios web generados en la fase *b*.
- d) Orquestación de servicios y generación de código. Los procesos de negocios definidos por la compañía describen cómo los servicios subyacentes deben coordinarse. Así, esta orquestación de servicios se obtiene mediante un proceso de tejido (*weaving*) de modelos utilizando como fuente los modelos BPMN, el modelo Simple-SoaML y el modelo BP-WS obtenido en la fase *c*, obteniéndose un nuevo modelo denominado Extended BPMN. Este modelo corresponde a una extensión de los modelos originales BPMN incluyendo las referencias concretas a los servicios alineados. En nuestro caso, estos nuevos modelos BPMN son ejecutables mediante Apache Activiti [1].

En este trabajo se toma como punto de partida las dos primeras fases de MigraSOA, por lo que se asume que el proceso de identificación de servicios desde la LWA ha sido completado [12]. A continuación, nos centraremos en las fases *c* y *d* de la Figura 1, en las que: i) se define un proceso de alineación de servicios utilizando un algoritmo semántico y ii) se realiza un proceso de orquestación de servicios y generación de código que permite inyectar información adicional en los modelos BPMN.

### 3 Alineación de los procesos de negocio

En esta sección se aborda la alineación de los procesos de negocio con los servicios web llevada a cabo en la fase *c*. La compañía debe definir en BPMN los procesos de negocio. Además, estos procesos de negocio deben definirse siguiendo las guías para la descripción de procesos de negocio operacionales definidas en [16], de forma que se aporte suficiente información sobre los mensajes entre tareas, incluyendo la información de parámetros de entrada y salida de las distintas tareas. En la figura 2 puede observarse un ejemplo de proceso de negocio sobre un sistema denominado CRS (*Conference Review System*).

La alineación automática de los procesos de negocio con los servicios web identificados se lleva a cabo utilizando un algoritmo semántico [13]. Este

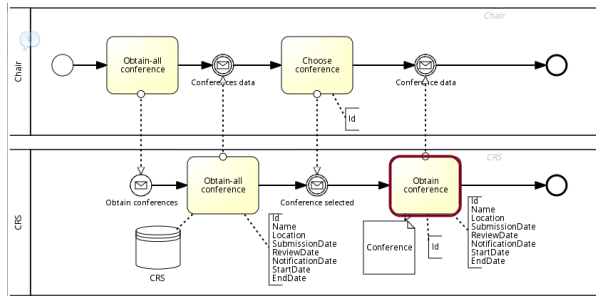


Fig. 2. Diagrama BPMN que representa el proceso de negocio *GetConference*

algoritmo ha sido adaptado para manejar la complejidad relacionada con la identificación de las tareas de los procesos de negocio y el análisis de los servicios web disponibles. Como ejemplo de identificación de las tareas definidas en los procesos de negocio, en la figura 2 se puede observar la tarea *Obtain conference* marcada con una línea más gruesa, sobre la cual se observan los atributos relacionados con la información general de una conferencia (*Id*, *Name*, *Location*, *SubmissionDate*, *ReviewDate*, *NotificationDate*, *StartDate* y *EndDate*). Para llevar a cabo la alineación se utiliza como base semántica un diccionario que representa el dominio de aplicación y se apoya fundamentalmente en la definición de un conjunto de términos y sus sinónimos para identificar posibles alineaciones semánticas entre las tareas de los procesos de negocio y los servicios web identificados en la LWA.

En Algoritmo 1 se describe el pseudocódigo correspondiente a la adaptación del algoritmo semántico de Wang-Ali [15], que originalmente fue propuesto para obtener una medida de similitud entre ontologías, de forma que cuanto más cercana a 1 era dicha medida las ontologías comparadas no tenían nada en común y cuanto más cercana a 0, más similares eran. Esta adaptación se ha realizado en dos partes: por una parte, el algoritmo original compara ontologías completas y en la versión adaptada se comparan dos conjuntos de términos: a) tareas y sus sinónimos con b) servicios y sus sinónimos (el conjunto de sinónimos de cada tarea o servicio son los términos relacionados según el diccionario semántico definido). Por otra parte, las etiquetas en las tareas y en los identificadores de los servicios están formadas por palabras compuestas, incluyendo una acción y un objeto, de modo que podemos identificar cada palabra y aplicar un peso configurable a cada una de ellas en la aplicación del algoritmo. Como ejemplo, la tarea *Obtain conference* mostrada en la figura 2 puede dividirse en dos palabras: *Obtain* (acción) y *conference* (objeto).

En el Algoritmo 1 se define  $\mathcal{T}$  como el conjunto de tareas de procesos de negocio y  $\mathcal{S}$  como el conjunto de servicios. De tal forma que, para cada tarea (llamada  $T$ ) en  $\mathcal{T}$ , se comprueba cada servicio (llamado  $S$ ) en  $\mathcal{S}$  y se calcula un valor de similitud entre cada tarea y cada servicio. Este valor estará en el rango [0-1], de modo que cuanto más cercano sea el valor a 0, más similares serán los términos comparados. Para diferenciar acción y objeto en cada término, nombraremos  $T1$  a la acción de la tarea  $T$  y  $T2$  al objeto de la tarea  $T$ . Lo mismo aplicamos en los servicios:  $S1$  será la acción del servicio  $S$  y  $S2$  será el objeto del mismo servicio. Por último, se aplican pesos a la acción y objeto que se denominan *weighta* y *weighto* respectivamente.

---

**Algoritmo 1** Algoritmo semántico para el proceso de alineación [12]

---

```

Wang-Ali-extended algorithm ( $\mathcal{T}$ ,  $\mathcal{S}$ , weighta, weighto, Wang-Ali-SimMatr[[]])
begin
for each  $T \in \mathcal{T}$  //for each task
     $T1 = \text{action of } T$ 
     $T2 = \text{object of } T$ 
     $Syn\_T1 = \text{set of synonyms of } T1$ 
     $Syn\_T2 = \text{set of synonyms of } T2$ 
for each  $S \in \mathcal{S}$  //for each service
     $S1 = \text{action of } S$ 
     $S2 = \text{object of } S$ 
     $Syn\_S1 = \text{set of synonyms of } S1$ 
     $Syn\_S2 = \text{set of synonyms of } S2$ 
     $v1 = Syn\_T1 - Syn\_S1 / Syn\_T1$ 
    //v1: semantic diff. between actions
     $v2 = Syn\_T2 - Syn\_S2 / Syn\_T2$ 
    //v2: semantic diff. between objects
     $Wang-Ali-SimMatr[T][S] = v1 * \text{weighta} +$ 
     $v2 * \text{weighto}$ 
    end for
end for
end

```

---

El resultado obtenido de este algoritmo corresponde a una matriz *Wang-Ali-SimMatr* de  $t$  filas (número de términos en el conjunto de tareas de procesos de negocio) y  $s$  columnas (número de términos en el conjunto de servicios). La  $i$ -ésima fila y  $j$ -ésima columna de la matriz contiene el valor *Wang-Ali-SimMatr* $_{ij} = Wang-Ali-SimMatr[t_i][s_j]$ , que representa el valor de similitud entre cada tarea definida en los procesos de negocio y cada servicio definido en la capa de servicios identificada en la fase  $b$  [12]. Como hemos indicado previamente, para cada *Wang-Ali-SimMatr* $[t_i][s_j]$  se obtiene un valor en el rango entre [0-1] que permite identificar la posible alineación entre una tarea y un servicio. Se ha establecido que sólo aquellas comparaciones que superen un umbral de 0.8 [13] (este umbral puede ser modificado por el ingeniero software) se consideran



alineaciones adecuadas. Téngase en cuenta que el cálculo del umbral *h* ha sido normalizado para representar valores comparables ascendentes:  $1 - \text{Wang-Ali-SimMatr}_{ij}$ .

El resultado obtenido en esta fase *c* incluye: i) tareas de procesos de negocio alineadas con los servicios subyacentes y ii) tareas de procesos de negocio que no han podido ser alineadas con ningún servicio. Ambas situaciones se recogen sobre un modelo denominado BP-WS (conforme con el metamodelo del mismo nombre, BP-WS) donde se modelan las relaciones de alineación obtenidas.

#### 4 Generación de la orquestación de los servicios mediante la extensión de los modelos BPMN

Los procesos de negocio describen cómo distintos servicios deben estar orquestados para alcanzar un determinado objetivo. En esta fase *d* partimos de los modelos BPMN, el modelo Simple-SOAML y el modelo que representa las tareas de procesos de negocio alineadas y los servicios web que deben implementarlas (modelo BP-WS). A partir de estas entradas se extienden los procesos de negocio (BPMN) con la información específica sobre la invocación a los servicios indentificados, obteniéndose así una nueva versión extendida de los modelos BPMN, los cuales serán ejecutables. En otras palabras, ya que en la fase *c* de MigraSOA se obtiene un modelo que identifica la alineación de tareas en los procesos de negocio y servicios, es necesario llevar esta información a los modelos concretos que las plataformas de ejecución de procesos de negocio requieren. En consecuencia, este proceso es específico de la plataforma concreta de ejecución de los procesos de negocio, en nuestro caso Apache Activiti [1].

La generación de código llevada a cabo permite: i) inyectar en los archivos .bpmn el código necesario para invocar los correspondientes servicios y ii) generar las clases Java que sirven de *wrapper* para invocar el servicio web requerido.

Por una parte, se debe localizar en el archivo .bpmn la información sobre la correspondiente tarea e inyectar la información sobre el servicio que debe ser invocado, incluyendo la dirección URL donde se localiza el WSDL del servicio web, el nombre de la operación que se va a invocar y el mapeo de parámetros de entrada y salida. Además, se debe definir la ubicación de una clase Java que será la encargada de llevar a cabo la invocación desde el motor de ejecución. Esta clase Java debe implementar la interface *org.activiti.engine.delegate.JavaDelegate*.

En este caso, la implementación de dicha interface se realiza utilizando Apache CXF [4], una implementación de JAX-WS <sup>1</sup>, que facilita la construcción de invocaciones sobre servicios web. Cabe destacar que la información desde el motor de ejecución hacia la clase Java definida es un parámetro de tipo *DelegateExecution execution*, a partir del cual podemos obtener la URL del WSDL, el nombre de la operación y la lista de parámetros de entrada y salida. En consecuencia, desde la implementación de esta clase podemos realizar la invocación al servicio.

<sup>1</sup> <https://jcp.org/en/jsr/detail?id=224>

Por lo tanto, la implementación concreta de esta interface permite: i) obtener una instancia dinámica del WS a partir de *JaxWsDynamicClientFactory* [4]; ii) crear un cliente para el WSDL del WS a invocar; iii) realizar la invocación pasando como argumentos la operación y un vector con los datos de entrada; y iv) obtener el resultado almacenado en un vector de datos de salida.

En la Figura 3 podemos observar el código XML necesario para extender una tarea en un proceso BPMN. Como puede observarse, será necesario vincular la tarea BPMN alineada con la clase que implementa la invocación del servicio Web (previamente descrita) y para ello se etiquetan los siguientes parámetros de la tarea: i) *TaskType* que identifica el tipo de tarea, específicamente *serviceTask*; ii) *activiti:class*, clase definida previamente que implementa la interface *org.activiti.engine.delegate.JavaDelegate* denominada *classes.WSDelegate*; iii) parámetros configurables, entre los que se añaden: *wSDL*, *operation*, *alignedParamsIn*, *alignedParamsOut* y *notAlignedParamsOut*. Los parámetros *alignedParamIn*, *alignedParamOut* y *notAlignedParamOut* forman parte del resultado obtenido en la fase anterior de MigraSOA (fase c), donde se alinean las tareas de los procesos de negocio con los servicios web que deben implementarlas.

```
<serviceTask id="serviceTask1" name="Name_of_the_Service_Tasks"
  activiti:class="classes.WSDelegate">
  <extensionElements>
  <activiti:field name="wSDL">
    <activiti:expression>URL where the WSDL web service
      is available</activiti:expression>
  </activiti:field>
  <activiti:field name="operation">
    <activiti:expression>operationName
    </activiti:expression>
  </activiti:field>
  <activiti:field name="alignedParamsIn">
    <activiti:expression>${paramIn1}
  </activiti:expression>
  </activiti:field>
  <activiti:field name="alignedParamsOut">
    <activiti:expression>${paramOut1}
  </activiti:expression>
  </activiti:field>
  <activiti:field name="notAlignedParamsOut">
    <activiti:expression>${notAlignedParamOut1}
  </activiti:expression>
  </activiti:field>
  </extensionElements>
</serviceTask>
```

Fig. 3. Ejemplo de código XML asociado a una tarea BPMN de tipo servicio

Como resultado en esta fase *d*, los modelos BPMN han sido extendidos mediante la inyección del código necesario para invocar adecuadamente los servicios que han sido alineados. En definitiva, se obtienen los procesos BPMN ejecutables adaptados al motor de procesos de negocio Apache Activiti. Cabe destacar que en los procesos BPMN pueden existir tareas que no han sido alineadas con servicios web, las cuales deben ser revisadas manualmente ya que podrían requerir la implementación adicional de algún servicio web que no está soportado por la LWA.

## 5 Conclusiones

En este trabajo se aborda, por una parte, la migración desde LWA hacia arquitecturas orientadas a servicios la cual puede realizarse utilizando técnicas de desarrollo de software dirigido por modelos que permiten manejar la complejidad de las plataformas tecnológicas actuales. Por otra parte, se permite a las empresas definir sus procesos de negocio, los cuales serán alineados con los servicios web ofrecidos por la LWA, facilitando de este modo la orquestación de los servicios en la arquitectura SOA. Además, se ha implementado un proceso de generación de código que permite extender la definición original de los procesos de negocios definidos en BPMN para inyectar el código necesario que permite la invocación de los servicios web alineados.

Actualmente se está trabajando en otros casos de estudio que incluyan procesos de negocios más complejos y nuevos generadores de código hacia otras plataformas de ejecución de procesos de negocio tales como Bonita Soft.

## Referencias

1. Activiti, A.: Apache Activiti, a Business Process Management (BPM) platform. <http://www.activiti.org> (2015), <http://www.activiti.org>
2. Almonaies, A., Cordy, J.R., Dean, T.R.: Legacy system evolution towards service-oriented architecture. In: International Workshop on SOA Migration and Evolution. pp. 53–62 (2010)
3. Almonaies, A.A., Alalfi, M.H., Cordy, J.R., Dean, T.R.: A framework for migrating web applications to web services. In: Daniel, F., Dolog, P., Li, Q. (eds.) Web Engineering - 13th International Conference, ICWE 2013, Aalborg, Denmark, July 8-12, 2013. Proceedings. Lecture Notes in Computer Science, vol. 7977, pp. 384–399. Springer (2013), <http://dx.doi.org/10.1007/978-3-642-39200-9>
4. Apache CXF: CXF User's Guide. Tech. rep. (2015), <http://cxf.apache.org/docs/>
5. Canfora, G., Fasolino, A.R., Frattolillo, G., Tramontana, P.: A wrapping approach for migrating legacy system interactive functionalities to Service Oriented Architectures. The Journal of Systems and Software 81(4), 463–480 (Apr 2008)
6. Davis, J.: Open Source Soa. Manning Publications Co., Greenwich, CT, USA, 1st edn. (2009)
7. Liu, Y., Wang, Q., Zhuang, M., Zhu, Y.: Reengineering legacy systems with restful web service. In: Computer Software and Applications, 2008. COMPSAC'08. 32nd Annual IEEE International. pp. 785–790. IEEE (2008)

8. O'Brien, Liam; Bass, L., Merson, P.F.: Quality attributes and service-oriented architectures. Software Engineering Institute. Paper 449. <http://repository.cmu.edu/sei/449/> (2005)
9. Rodríguez-Echeverría, R., Conejero, J.M., Linaje, M., Preciado, J.C., Sánchez-Figueroa, F.: Re-engineering legacy Web applications into Rich Internet Applications. In: 10th International Conference on Web Engineering. vol. 6189, pp. 189–203. LNCS, Springer (2010)
10. Sneed, H.M.: Integrating legacy software into a service oriented architecture. In: CSMR. pp. 3–14. IEEE Computer Society (2006)
11. SOA-Manifesto: <http://www.soa-manifesto.org> (2011)
12. Sosa, E., Clemente, P.J., Conejero, J.M., Rodríguez-Echeverría, R.: A model-driven process to modernize legacy web applications based on service oriented architectures. In: WSE. pp. 61–70 (2013)
13. Sosa, E., Clemente, P.J., Sánchez-Cabrera, M., Conejero, J.M., Rodríguez-Echeverría, R., Sanchez-Figueroa, F.: Service discovery using a semantic algorithm in a soa modernization process from legacy web applications. In: IEEE Tenth World Congress On Services. SERVICES. IEEE Computer Society (2014)
14. Ullah, A., Lai, R.: A systematic review of business and information technology alignment. ACM Transactions on Management Information Systems (TMIS) 4(1), 4 (2013)
15. Wang, J.Z., Ali, F., Appanavanda, R.: A web service for efficient ontology comparison. In: Proceedings of the IEEE International Conference on Web Services. pp. 843–844. ICWS '05, IEEE Computer Society, Washington, DC, USA (2005)
16. Weske, M.: Business Process Management - Concepts, Languages, Architectures, 2nd Edition. Springer (2012), <http://dx.doi.org/10.1007/978-3-642-28616-2>
17. Zhang, B., Bao, L., Zhou, R., Hu, S., Chen, P.: A black-box strategy to migrate gui-based legacy systems to web services. In: Service-Oriented System Engineering, 2008. SOSE'08. IEEE International Symposium on. pp. 25–31. IEEE (2008)

# SOA 2.0 para el Control y Alerta de Riesgos para la Salud Derivados de la Calidad del Aire

Alfonso García de Prado, Guadalupe Ortiz, Juan Boubeta-Puig

Grupo UCASE de Ingeniería del Software  
Escuela Superior de Ingeniería, Universidad de Cádiz,  
Avda. de la Universidad de Cádiz nº 10, 11519 Puerto Real, Cádiz, España

{alfonso.garciadeprado, guadalupe.ortiz, juan.boubeta  
}@uca.es

Adolfo R. de Soto

Dpto. de Ingenierías Mecánica, Informática y Aeroespacial  
Universidad de León, Campus de Vegazana, s/n, 24071 León

adolfo.rdesoto@unileon.es

**Resumen.** La calidad del aire es un factor al que se la ha dado gran relevancia en los últimos años dado que puede afectar seriamente a la salud y a la calidad de vida de las personas. En la actualidad carecemos de medios que nos proporcionen información sobre la calidad del aire en tiempo real y de fácil acceso para los ciudadanos; y en especial no hay medios que se puedan particularizar para las condiciones específicas de cada individuo. En este artículo mostramos nuestros avances en la propuesta, implementación y prueba de una arquitectura orientada a servicios y dirigida por eventos que nos permite detectar en tiempo real los cambios en la calidad del aire y ponerlos al alcance de los ciudadanos enviándoles notificaciones y alertas personalizadas en función de sus características personales, previniendo así mayores riesgos para la salud.

**Keywords:** Arquitecturas Orientadas a Servicios, Arquitecturas Dirigidas por Eventos, Procesamiento de Eventos Complejos, Bus de Servicios Empresariales, Calidad del Aire.

## 1 Introducción

La polución afecta de forma vehemente a la salud de las personas: como ya vienen haciéndose eco los medios de telecomunicación, la calidad del aire juega un papel de gran relevancia cuando se trata de la salud de los ciudadanos, pudiendo propiciar o empeorar determinadas enfermedades o incluso causar la muerte en determinados grupos de riesgo (personas con enfermedades respiratorias o cardiovasculares, entre otros). En general todos podemos vernos afectados por la mala calidad del aire.

La calidad del aire cambia constantemente; es por esto que es importante estar informado de esta en tiempo real, pudiendo así tomar medidas en consecuencia. En este sentido, sería muy deseable no sólo que la información se actualizase en tiempo real, sino que pudiese llegar al ciudadano, de forma fácil e intuitiva, también en tiempo real. Además, lo deseable sería también que la información estuviese personalizada para el usuario en función de sus características y dolencias particulares. Este hecho ya lo destacábamos en estas mismas jornadas hace 2 ediciones [1], donde plasmábamos la arquitectura inicialmente planteada para este sistema. Se esperaba que dicha infraestructura software fuese capaz de procesar diversas fuentes de información que nos permitiesen conocer y valorar la calidad del aire en cada instante, detectando las correspondientes alertas por mala calidad del aire y, finalmente, gestionando la información personal del usuario, así como las alertas que le serían enviadas.

En este artículo mostramos los avances realizados en esta línea, centrándonos en la parte servidora. El objetivo principal de este trabajo ha sido propiciar la adición de una serie de motas<sup>1</sup> de calidad del aire con diversos formatos y sensores sin afectar al funcionamiento del sistema, así como dotar al sistema de una serie de servicios y bases de datos que permitan la personalización de las alertas para los futuros clientes del sistema. De este modo se crea una arquitectura que puede ser usada por clientes software orientados tanto al ciudadano de a pie, como a los servicios sanitarios como, por ejemplo, a las administraciones o agencias de protección medioambiental; siendo para todos ellos transparente la arquitectura hardware y software y haciéndosela fácilmente accesible a través de servicios REST.

El resto del artículo se organiza de la siguiente manera: en la sección 2 se introducen conceptos preliminares tecnológicos y relacionados con la calidad del aire. A continuación, la sección 3 enumera los objetivos del sistema y la 4 describe la arquitectura implementada para dicho sistema. Después, en la sección 5 se explican qué patrones se han implementado y la sección 6 describe cómo se ha probado el sistema. Por últimos, las secciones 7, 8 y 9 muestran la discusión del sistema, los trabajos relacionados y las conclusiones y trabajo futuro, respectivamente.

## 2 Conceptos Previos

A continuación, se introducen conceptos previos sobre arquitecturas orientadas a servicios y dirigidas por eventos (*Event-Driven Service-Oriented Architecture*, SOA 2.0), procesamiento de eventos complejos (*Complex Event Processing*, CEP), Internet de las cosas (*Internet of Things*, IoT) y calidad del aire.

### 2.1 Arquitecturas orientadas a servicios y dirigidas por eventos

Una SOA 2.0 es una evolución de una SOA tradicional en la que las comunicaciones entre usuarios y servicios se lleva a cabo a través de eventos, en lugar de mediante

---

<sup>1</sup> Se utiliza el término *mota* para referirse a cada conjunto de sensores presentes en una determinada localización, ya sean propios o de terceros.

llamadas a procedimientos remotos [2]. Para lograr esta integración, se requiere una capa de abstracción de software que integre diversas fuentes de datos heterogéneas y que permita también distribuir invocaciones a diversos componentes software [3]. Estas funcionalidades son ofrecidas por un bus de servicios empresariales (*Enterprise Service Bus*, ESB), que posibilita la interoperabilidad entre protocolos de comunicación diferentes y fuentes y destinos de datos heterogéneos.

A pesar de todas las ventajas que proporciona una arquitectura orientada a servicios y dirigida por eventos, este tipo de arquitectura no es adecuada para procesar, analizar y correlacionar una ingente cantidad de datos, en forma de eventos, con el fin de detectar en tiempo real situaciones relevantes para un dominio particular. Para cubrir esta necesidad es necesario integrar procesamiento de eventos complejos.

## 2.2 Procesamiento de eventos complejos

CEP es una tecnología que permite capturar, analizar y correlacionar una ingente cantidad de eventos heterogéneos con el fin de detectar situaciones críticas o relevantes en tiempo real para un determinado dominio de aplicación. Se basa en el filtrado de eventos irrelevantes y en el reconocimiento de los que sí son relevantes para dicho dominio. Para ello, se utilizan unos patrones de eventos en los que se especifican las condiciones que deben cumplirse para detectar dichas situaciones; a estas situaciones de una mayor complejidad semántica se les denominan eventos complejos [2]. La característica principal de estos eventos complejos procesados mediante tecnología CEP es que pueden ser identificados e informados en tiempo real, reduciendo así la latencia en la toma de decisiones, a diferencia del software tradicional de análisis de eventos que no funciona en tiempo real.

## 2.3 Internet de las Cosas

IoT se define como una red formada por objetos interconectados e identificados unívocamente [4]; en la que podemos encontrar elementos para diversos dominios de aplicación. Existen múltiples plataformas de Internet de las Cosas de donde obtener datos en tiempo real. Una de ellas es ThingSpeak, una plataforma de uso gratuito, bien documentada y altamente escalable, que ofrece una API REST para la gestión de los datos en múltiples formatos. El objetivo de este proyecto no es centrarse en una determinada plataforma de Internet de las Cosas, sino poder integrar cualquier plataforma o mota de tercero; se ha usado ThingSpeak a modo de ilustración.

## 2.4 Calidad del Aire

Para los valores de referencia de calidad del aire hemos seguido los datos proporcionados por la Agencia de Protección Medioambiental de los EE.UU. (EPA) [5] [6].

En estos documentos podemos encontrar la denominación mostrada en la Tabla 1 sobre la bondad de la calidad del aire con carácter general en función del parámetro calculado para el índice de calidad del aire (*Air Quality Index*, AQI), la influencia en general a la población, así como las recomendaciones para ésta. Según la EPA, el

nivel AQI de calidad del aire será el mayor de los niveles de calidad del aire de cada uno de los contaminantes de este; es decir, si el O<sub>3</sub>, el CO, el NO<sub>2</sub>, EL PM<sub>2,5</sub> y el PM<sub>10</sub> están, por ejemplo, en niveles inferiores al 4 (los hemos numerado para mayor comodidad); pero el SO<sub>2</sub> está en 4; entonces se considera que el nivel de calidad del aire en general es 4 (Insalubre). En la Tabla solo mostramos 2 niveles, de los 6 existentes, por restricciones de espacio.

**Tabla 1.** Categorización de calidad del aire AQI.

Valor AQI	Calidad del Aire	Influencias	Recomendaciones
Nivel 2 (51 a 100)	Moderada	La polución y el ozono pueden afectar muy levemente a personas inusualmente sensibles a síntomas respiratorios.	Personas inusualmente sensibles deberían considerar reducir esfuerzos intensos o prolongados en el exterior.
Nivel 3 (101 a 150)	Insalubre para grupos sensibles	Las personas con enfermedades pulmonares o cardíacas, niños, ancianos y personas que realicen actividades en el exterior podrían verse afectadas.	Las personas con enfermedades pulmonares, como asma, los niños, ancianos y personas que realicen actividades en el exterior deberían reducir esfuerzos intensos o prolongados en el exterior.

Limitarnos al índice general de AQI sería una aproximación bastante pobre; por lo que proponemos llevar un control de cada uno de los contaminantes del aire cuya concentración es relevante para la salud. Dependiendo de la concentración para cada uno de ellos la salud de los ciudadanos se ve afectada de distinto modo. En la Tabla 2, detallamos, a modo de ejemplo, las condiciones que la EPA considera relevante para los valores de Ozono en periodos de 8 horas para 2 niveles y cómo afectan a distintos grupos de riesgo y a la población en general [6].

**Tabla 2.** Referencia AQI para los valores de Ozono en periodos de 8 horas.

Categorías AQI	Ozono (ppm) [8 horas]	Valores e Influencias
Nivel 2 - Moderada (51-100)	Valores	0.060-0.075
	Influencias	Personas inusualmente sensibles podrían experimentar síntomas respiratorios.
Nivel 3- Insalubre para grupos sensibles (101-150)	Valores	0.076-0.095
	Influencias	Posible incremento de síntomas respiratorios e incomodidad para respirar en personas con afecciones pulmonares, como asmáticos, niños, ancianos y personas que trabajen en el exterior.

### 3 Requisitos del sistema

El requisito principal del sistema es poder procesar diversas fuentes de información que nos permitan conocer y valorar la calidad del aire en cada instante; debe así mis-



mo detectar las correspondientes alertas por mala calidad del aire y, finalmente, debe ser capaz de enviar alertas personalizadas al usuario. Para ello será necesario:

1. Que el sistema pueda leer fuentes de datos heterogéneos de concentraciones de los contaminantes del aire; y que el conjunto de datos sea ampliable y personalizable.
2. Que nuestro sistema disponga de información actualizada de cuáles son los niveles de calidad del aire perjudiciales en mayor o menor medida para la salud en general y para determinados grupos de riesgos en particular.
3. Que nuestro sistema detecte en tiempo real, a partir de los datos de concentración del aire de entrada, los niveles de calidad del aire y que genere las consecuentes alertas, también de manera inmediata.
4. Que nuestro sistema permita el registro de usuarios que estén interesados en conocer la calidad del aire en tiempo real y ser alertados en caso de que haya algún peligro para su salud.
5. Que nuestro sistema pueda notificar las alertas personalizadas a los usuarios en función de sus características particulares.
6. Que nuestro sistema permita consultar los históricos de las concentraciones de los contaminantes del aire con cualquier fin que sea necesario (estadísticos, médicos, medioambientales, etc.).

#### 4 Arquitectura del sistema

La arquitectura propuesta e implementada, representada en la Fig. 1, está compuesta de los siguientes elementos:

- ESB que va a canalizar todas las comunicaciones: detección de eventos simples (datos de concentración de los contaminantes del aire en las motas), transformación y encaminamiento de estos al motor CEP y a la base de datos, detección de eventos complejos notificados por el motor CEP, consultas a las bases de datos de usuarios y envío de notificaciones a estos en función de sus suscripciones. Es importante destacar que se realizan diversas transformaciones de datos en el ESB (*data transform* en la figura de la arquitectura), estas transformaciones permiten obtener datos de fuentes heterogéneas y adaptarlos a un formato común que es enviado a las bases de datos y al motor de procesamiento de eventos complejos.
- Base de datos de concentraciones de contaminantes del aire en las motas.
- Servicio web REST que permite la consulta de los datos de concentraciones de contaminantes del aire en las motas. Queremos resaltar que se ha optado por el uso de servicios REST por su sencillez y fácil integración con cualquier futuro cliente que se quiera implementar para el sistema.
- Base de datos personales de usuarios y sus características particulares y también de datos de las distintas alertas disponibles en el sistema.
- Servicio web REST que permite la consulta de las alertas disponibles y el almacenamiento de datos personales de los usuarios y las alertas a las que se suscriben.
- Motor CEP que analiza los datos que provienen de las motas. En dicho motor se han desplegado los patrones de las alertas de calidad del aire que nos permiten de-

techar cuándo se dan dichas situaciones de alerta. Se ha implementado un conector para la integración del motor CEP con el ESB.

- Proveedor de datos (concentraciones de contaminantes del aire) de Internet de las Cosas en la nube (ThingSpeak). Desde el ESB se hacen consultas periódicas a la API ofrecida por ThingSpeak para la obtención de los datos.
- Sensores de terceros y sensores propios (simulador, en este caso) que proporcionan datos de concentraciones de los contaminantes del aire (ozono (O3), partículas de polvo (PM2.5 y PM10), monóxido de carbono (CO), dióxido de carbono (CO2), dióxido de sulfato (SO2), dióxido de nitrógeno (NO2) y temperatura y humedad). Aunque la temperatura y humedad no se han utilizado en los patrones del sistema, se han incluido dado que pueden ser de utilidad en el futuro y casi todas las placas de calidad del aire contienen sensores de este tipo.

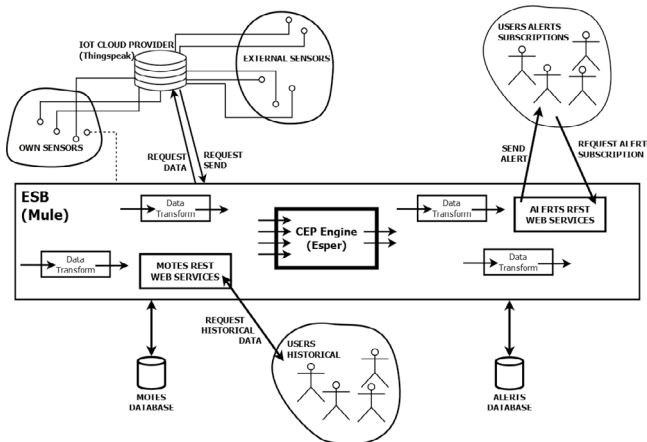


Fig. 1. Arquitectura del sistema.

A continuación se indican qué elementos de la arquitectura están implicados en el cumplimiento de los requisitos del sistema:

- El requisito 1 se consigue a través de la conexión del ESB con ThingSpeak y la definición y uso de la base de datos de las motas.
- El requisito 2 se consigue a través de la conexión del ESB con ThingSpeak y la definición de los patrones de calidad del aire en el motor CEP.
- El requisito 3 se consigue a través de la definición de los patrones en el motor CEP y la integración de éste último con el ESB y las bases de datos donde se definen las alertas.

- Los requisitos 4 y 5 se consigue mediante la oferta del servicio REST de usuarios y alertas, su integración con las bases de datos del sistema y con el bus de servicios empresariales.
- El requisito 6 se consigue mediante la integración de ThingSpeak, el ESB y las bases de datos, así como el servicio REST de las motas

## 5 Patrones de calidad del aire

Para la definición de los patrones de calidad del aire se ha seguido el índice de calidad del aire de la EPA [5] [7].

Se han definido patrones para cada uno de los niveles en los que había algún tipo de influencia para la salud y se han desplegado en el motor CEP. Por ejemplo, para el Ozono en períodos de 8 horas se han definido los patrones que detectan Nivel 2, Nivel 3, Nivel 4, Nivel 5 y Nivel 6; con los rangos indicados en cada uno de ellos (ver niveles 2 y 3 en la Tabla 3). Para cada contaminante se ha definido desde el Nivel 2 o 3 hasta el 6; pueden consultarse en detalle en el documento de la EPA [5].

**Tabla 3.** Rangos de valores para los 2 patrones definidos para Ozono en períodos de 8 horas.

Categorías AQI	Ozono (ppm) [8 horas]
Nivel 2 - Moderada (51-100)	0.060-0.075
Nivel 3- Insalubre para grupos sensibles (101-150)	0.076-0.095

## 6 Prueba del Sistema

Se ha creado un simulador para generar datos para una serie de motas y sensores de modo que podamos probar exhaustivamente el sistema (de otro modo solo podríamos hacer pruebas con valores actuales públicos de calidad del aire, que no necesariamente toman valores de riesgo) y se han enviado dichos datos a ThingSpeak, como si de datos de terceros se tratase. A continuación se destacan algunas de las características generales del simulador:

- El simulador al arrancar nos permite añadir tantas motas como queramos.
- Cada nueva mota se crea con una serie de sensores predefinidos que pueden modificarse en cualquier momento.
- Pueden salvarse o cargarse configuraciones previamente salvadas.
- En cualquier momento se puede activar o desactivar el envío de datos, así como editar a mano los valores de cada sensor (por si se quieren probar unos valores concretos y no aleatorios).
- La frecuencia de muestreo, los rangos de valores de los sensores, el número de pasos entre el mínimo y el máximo, la suavidad de las curvas simuladas, así como la probabilidad de romper la tendencia de una curva son totalmente configurables.

Usando este simulador hemos configurado diversas motas y verificado que cuando se dan situaciones de mala calidad del aire, se detectan los consecuentes valores de riesgo y se lanzan las correspondientes alertas, así como las notificaciones a los usuarios interesados en estas.

Es importante destacar que el simulador se ha probado con distintas cadencias en el tiempo de envío de datos y que la arquitectura propuesta es absolutamente configurable en la cadencia para los tiempos de lectura de los datos. En el momento que un evento (una lectura de datos) entra en el sistema, esta ya se procesa en tiempo real sin ningún tipo de limitación temporal; así pues podemos detectar los cambios de calidad del aire en tiempo real. No obstante, consideramos que una cadencia de 5 o 10 minutos (en la lectura de datos) es más que suficiente para este campo de aplicación.

## 7 Discusión

Se quieren resaltar los siguientes aspectos positivos de la implementación:

1. Se utilizan los valores de referencia de calidad del aire proporcionados por la agencia medioambiental de Estados Unidos por estar estandarizados y porque detallan claramente a qué personas afectan en mayor o menor medida y cuáles son las recomendaciones para éstas. Por supuesto, sería fácilmente aplicable para otros valores de referencia, los Europeos por ejemplo y perfectamente aplicables en España. Para ello sólo habría que desplegar nuevos patrones en el sistema.
2. Se permite identificar a los sensores con una serie de sinónimos; por ejemplo, la temperatura podría encontrarse con algunos de los siguientes identificadores: *temperatura*, *temperature*, *temp*, *t*, o cualquier otro elegido por el usuario. El sistema es capaz de manejar todos estos datos como si de un identificador único se tratase, de forma transparente para el proveedor de datos y para los usuarios finales. Esto nos va a facilitar el procesamiento de diversas fuentes de información.
3. El sistema es escalable y configurable:
  - El sistema es escalable en tanto en cuanto está preparado para poder funcionar con tantas motas como se añadan en la base de datos; solo es necesario conocer (a través de la base de datos) su IP y los sensores de que dispone. Esto se hace de forma totalmente automática.
  - Así mismo es escalable en cuestión de sinónimos; esto es, se pueden añadir motas de terceros que utilicen diferentes identificadores para los contaminantes del aire, sin necesidad de modificar la implementación del sistema, que será capaz de procesarlos igualmente (tal como se ha explicado en el punto 1 de esta sección). Como ya hemos mencionado, esto nos va a facilitar el procesamiento de diversas fuentes de información.
  - El sistema es configurable en cada lectura de datos; ya que no necesariamente hay que obtener información de todos los sensores en cada lectura, sólo de aquéllos que tengan información nueva; siendo por tanto más eficiente.
  - En la base de datos se configura también en qué sensores estamos interesados actualmente (puede haber sensores de ciertas motas que no sean relevantes en un momento dado), así como se especifica el orden en el que vamos a

leer de los sensores, ya que la definición de la estructura con los eventos que se van a enviar al motor Esper se realiza automáticamente con estos datos.

Por otra parte, queremos también mencionar las siguientes limitaciones:

1. El sistema implementado únicamente envía notificaciones vía email, aunque estamos trabajando en el envío de notificaciones móviles y otras alternativas de comunicación que faciliten el acceso al ciudadano.
2. En la actualidad el sistema se basa en datos estáticos del usuario (enfermedades, edad, etc.) para avisar de las posibles alertas relacionadas con la calidad del aire; nuestro trabajo en curso afronta también cómo ampliar el sistema con datos dinámicos (ejercicio al aire libre, trabajos que requieran esfuerzos, etc.) para particularizar aún más las alertas a la situación concreta del ciudadano.
3. Finalmente, aunque trabajamos con simuladores y un limitado número de sensores; en el futuro esperamos poder trabajar con un amplio número de sensores de terceras partes que nos permitan disponer de esta información en diversas localizaciones geográficas para poder dar un mayor soporte. Actualmente contamos con el soporte de la Junta de Andalucía que nos va a proporcionar datos en tiempo real para poder realizar pruebas reales del sistema.

## 8 Trabajos Relacionados

Existen diversos sistemas para la monitorización de la calidad del aire específicos para determinadas áreas geográficas. Uno de los más conocidos es EnviroFlash [8], que permite calcular el AQI cada hora en los Estados Unidos. Para obtener los datos hay que hacerlo a través de la visualización de la web. De forma similar, en Europa disponemos de AirQualityNow [9], que calcula el *Common Air Quality Index* también cada hora para algunas ciudades específicas de Europa. De forma análoga, el gobierno de Canadá ha desarrollado también una herramienta web para monitorizar cada hora su propio *Air Quality Health Index* [10]. Para obtener los datos hay que hacerlo clicando en cada región del mapa y no tienen una interfaz para la integración con terceros. Un último ejemplo es el *Department for Environment Food & Rural Affairs* de Reino Unido, que mantiene una herramienta web llamada UK-Air [11] para monitorizar la calidad del aire en su país. De nuevo, hay que chequear la calidad del aire en el mapa clicando en cada localidad.

En general, no hemos encontrado ningún sistema que satisfaga los requisitos del sistema aquí propuesto: que la información se actualice en tiempo real (al minuto), que se pueda acceder a ella de forma fácil e intuitiva, y que las notificaciones y la información se pueda personalizar en función de las características del usuario.

## 9 Conclusiones y Trabajo Futuro

En este trabajo se ha demostrado la viabilidad de un sistema basado en arquitecturas orientadas a servicios y dirigidas por eventos para la prevención de riesgos para la salud derivados de la mala calidad del aire, mediante el envío de alertas personalizadas en función de las características personales del usuario registrado en el sistema y

de los niveles de calidad del aire en tiempo real. El sistema se ha diseñado de modo que sea escalable y altamente configurable, de modo que pueda integrarse con un número indefinido de motas y sensores de diversas partes sin necesidad de modificar la estructura del sistema, y que este pueda seguir funcionando normalmente.

En el futuro esperamos a) adaptar las notificaciones para que lleguen de forma más cómoda al ciudadano, b) integrar un mayor número de motas y sensores para cubrir diversas zonas geográficas, así como proporcionar mecanismos para la fácil configuración de los patrones en función de la zona geográfica y c) adaptar las notificaciones a la actividad concreta que el ciudadano registrado en el sistema esté realizando en cada momento.

## Agradecimientos

Los autores de la Universidad de Cádiz agradecen la financiación del Ministerio de Economía y Competitividad a través de los proyectos TIN2015-65845-C3-3-R y TIN2014-53986-REDT y el proyecto de investigación-UCA PR2016-032; también agradecen el interés y el apoyo de las empresas 4gotas.com y Novayre Solutions.

## Referencias

1. Boubeta-Puig, J., Ortiz, G., Medina-Bulo, I.: Una Propuesta Orientada a Servicios para la Prevención de Riesgos Personales Derivados de la Calidad del Aire. In: Actas de las X Jornadas de Ciencia e Ingeniería de Servicios. pp. 58–67, Cádiz, Spain (2014).
2. Luckham, D.C.: The power of events: an introduction to complex event processing in distributed enterprise systems. Addison-Wesley (2002).
3. Papazoglou, M.P., Heuvel, W.V.D.: Service-oriented design and development methodology. *Int J Web Eng Technol.* 2, 412–442 (2006).
4. Atzori, L., Iera, A., Morabito, G.: The Internet of Things: A survey. *Comput. Netw.* 54, 2787–2805 (2010).
5. U.S. Environmental Protection Agency: AQI Air Quality Index. A Guide to Air Quality and Your Health, [http://www3.epa.gov/airnow/aqi\\_brochure\\_02\\_14.pdf](http://www3.epa.gov/airnow/aqi_brochure_02_14.pdf).
6. U.S. Environmental Protection Agency, Research Triangle Park, North Carolina: Technical Assistance Document for the Reporting of Daily Air Quality – the Air Quality Index (AQI), <http://www3.epa.gov/airnow/aqi-technical-assistance-document-dec2013.pdf>.
7. U.S. Environmental Protection Agency: Air Quality Index (AQI) Basics, <http://airnow.gov/index.cfm?action=aqibasics.aqi>, (2016).
8. EPA - US: EnviroFlash, <http://www.enviroflash.info/>.
9. CITEAIR: Air Quality Now in Europe, <http://airqualitynow.eu/>.
10. Government of Canada: National Map - Air - Environment Canada, <http://www.ec.gc.ca/cas-aqhi/default.asp?lang=En&n=450C1129-1>.
11. Department for Environment Food & Rural Affairs - UK: UK-AIR, air quality information resource, <http://uk-air.defra.gov.uk/>.

# El factor humano en los procesos de negocio





# Recompilación de procesos de educación a partir de registros de eventos\*

Juan C. Vidal, Borja Vázquez-Barreiros, Manuel Lama, and Manuel Mucientes

Centro Singular de Investigación en TecnoloXías da Información (CiTIUS)  
Universidade de Santiago de Compostela, Spain  
{juan.vidal, borja.vazquez, manuel.lama, manuel.mucientes}@usc.es

## 1. Resumen

Existen dos factores principales que dificultan la realización de un escenario educativo a la hora de diseñar un curso: *i*) cómo modelar un escenario pedagógico práctico que permita conseguir los objetivos educativos, y *ii*) cómo reusar este escenario en un contexto que no sea el original. Generalmente, los profesores no definen únicamente el contenido que deben de consumir los alumnos durante un curso, si no que también definen los objetivos pedagógicos, en qué orden se deben realizar las diferentes actividades, el sistema de evaluación, etc. Por lo tanto, para poder reutilizar y reusar un curso, es necesario que este esté explícitamente detallado. Aunque en general el diseño de los cursos suelen estar documentados utilizando lenguaje natural, su descripción se puede formalizar utilizando un Lenguaje de Modelado Educativo (EML). Típicamente, en un entorno de aprendizaje virtual, los alumnos pueden realizar actividades adicionales de las explícitamente definidas por el profesor, como publicar en un foro o revisar la bibliografía. Por lo tanto, esta información se debería de resaltar para que los profesores puedan mejorar tanto el proceso de evaluación, así como el *flujo educativo*, es decir, el proceso real seguido por los alumnos. De este modo, un escenario educativo es mucho más complejo que el explícitamente diseñado por los profesores.

La principal idea detrás de un EML es que permiten describir, desde un punto de vista pedagógico, el proceso de aprendizaje de un curso, es decir, la secuencia de actividades que un alumno debe de seguir para adquirir los los objetivos educativos del curso. Dentro de la gran variedad de especificaciones para representar diseños educativos, IMS LD es el estándar *de facto* dentro del panorama educativo. Este estándar permite describir procesos educativos en un entorno de aprendizaje virtual. Sin embargo, muchos autores consideran que algunas de las características de este estándar son demasiado complejas, lo que dificulta su completa adopción por parte de los profesores. Para hacer frente a esta problemática, se han desarrollado diferentes herramientas para facilitar la creación de diseños educativos. Sin embargo, incluso con estas herramientas, el

---

\* Vidal, J.C., Vázquez-Barreiros, B., Lama, M., Mucientes, M.: Recompiling learning processes from event logs. *Knowledge-Based Systems* **100** (2016) 160–174

proceso de creación de un curso en IMS LD estándar es complejo y se complica cuando se quieren utilizar componentes avanzadas de este estándar.

La reconstrucción automática de este tipo de cursos podría aliviar esta problemática, estimulando el uso de IMS LD entre profesores. Tomando como punto de partida el registro de eventos que almacena cada una de las acciones realizadas por los alumnos en un entorno de aprendizaje virtual, es posible descubrir el *comportamiento real* seguido por los alumnos durante un curso, es decir, qué es lo que hicieron en verdad los alumnos, así como las diferentes restricciones en el proceso de aprendizaje. Así, combinando estos dos modelos, es posible reconstruir un curso en un EML específico. De este modo, es posible la reutilización de cursos independientemente del entorno de aprendizaje virtual utilizado. Por lo tanto, los profesores pueden definir un curso dentro de cualquier entorno de aprendizaje virtual, evitando el tener que utilizar una herramienta de diseño en un EML específico, y aún así ser posible la reconstrucción del curso desde cero y transformarlo a, por ejemplo, IMS LD.

En este artículo [1] se presenta un *proceso de reingeniería* que permite reconstruir, de manera automática, un curso en el estándar IMS LD, tomando como punto de partida los eventos generados por los alumnos en un entorno de aprendizaje virtual. Primero, utilizando un algoritmo de minería de procesos, se descubre de manera automática el flujo educativo a partir de las secuencias de actividades seguidas por los alumnos. A continuación, utilizando conocimiento del dominio, un algoritmo determina qué componentes de IMS LD se deben de crear. Finalmente, a partir de los valores de las variables del registro de evento, se extraen automáticamente las reglas adaptativas del curso —utilizando un árbol de decisión—, y se agregan a la estructura de IMS LD previamente descubierta. Las principales contribuciones de este proceso de reingeniería son: *i*) una nueva infraestructura que facilita la reutilización de cursos entre diferentes entornos de aprendizaje virtual; *ii*) el descubrimiento automático de procesos de aprendizaje a partir de registros de eventos y su recompilación a IMS LD; y *iii*) la identificación automática de reglas adaptativas, representadas como árboles de decisión, a partir de registro de eventos.

Respecto a la validación, el presente proceso de reingeniería se ha validado utilizando nueve cursos con diferentes niveles de complejidad, obteniendo en todos los casos la estructura IMS LD correcta. Además, el presente sistema sólo requiere de la participación de los profesores a la hora de seleccionar cuál es la solución que mejor describe los objetivos pedagógicos del curso. Así, desde el punto de vista de los profesores, se facilita el reuso de cursos educativos entre diferentes entornos de aprendizaje.

## Referencias

1. Vidal, J.C., Vázquez-Barreiros, B., Lama, M., Mucientes, M.: Recompiling learning processes from event logs. *Knowledge-Based Systems* **100** (2016) 160–174

# BPMS-Game: Herramienta para la Gamificación de Procesos de Negocio

Javier Mancebo, Félix García

Instituto de Tecnologías y Sistemas de Información, Universidad de Castilla la Mancha, Ciudad Real, España

{javier.mancebo | felix.garcia}@uclm.es

**Resumen.** En los últimos años, el paradigma BPM está teniendo una repercusión significativa en las organizaciones a la hora de dar soporte a la mejora continua de sus procesos de negocio. Uno de los aspectos que puede promover un mejor desempeño de dichos procesos es el factor humano, por lo que es de gran importancia mejorar la motivación de los usuarios para la realización de las tareas que tienen encomendadas. En este contexto, resulta de interés el área de “Gamificación”, que está muy presente en muchos aspectos de la sociedad actual, con el objetivo de mejorar la participación y el compromiso de un usuario para la realización de sus tareas aplicando mecanismos de juegos. La gamificación puede ser por tanto un mecanismo adecuado para su aplicación en los procesos de las organizaciones para mejorar la motivación y el rendimiento de los usuarios involucrados. Por todo ello, en el presente artículo se describe la herramienta BPMS-Game, que da soporte a la aplicación de gamificación en sistemas BPMS. La utilidad potencial de la herramienta se ilustra con un caso de ejemplo.

**Palabras clave:** Procesos de Negocio, Gamificación, BPMS.

## 1 Introducción

En el presente artículo se abordan los posibles beneficios que se pueden obtener con la aplicación de gamificación en un entorno empresarial, especialmente para las empresas u organizaciones que han ido evolucionando en los últimos años hacia la gestión de procesos de negocio (BPM, Business Process Management). BPM es un paradigma que centra sus esfuerzos en la optimización de los procesos de negocio de la organización, buscando de esta manera la mejora de la productividad, la eficacia y la eficiencia de la gestión sistemática de los procesos que deben ser modelados, automatizados, integrados, monitorizados y optimizados de forma continua [1]. La idea clave de BPM es centrarse en los procesos y la gestión de trabajo en una organización. Los componentes sobre los que se basa BPM son los procesos, las personas y la tecnología [2]. Mediante BPM se aportan diversos beneficios a las organizaciones, como una mayor flexibilidad y agilidad para adaptarse al cambio; mejora de la eficiencia, eliminando pérdidas de tiempo, con la consiguiente reducción de costes. Otra de las principales razones para la implantación de BPM es que mejora la

satisfacción de los clientes, ya que lo procesos se ejecutan de forma más rápida, eficiente y efectiva, permitiendo que los clientes reciban mejores productos, en menor tiempo y con una mejor experiencia [1].

Entre los factores de éxito de una adecuada implantación del paradigma BPM en la organización se encuentra el factor humano [3, 4]. El paradigma BPM ha evolucionado mucho en los últimos años y se han conseguido importantes avances en muchos aspectos como es la estandarización, automatización, evaluación y mejora, pero también es fundamental considerar la influencia que las personas y sus interacciones pueden tener en el éxito de una empresa. En este sentido cabe destacar el potencial que puede tener una adecuada aplicación del concepto de Gamificación (del inglés, *gamification*), que ha experimentado un gran crecimiento en los últimos años. La gamificación se puede definir como el “uso de elementos de diseño de juegos en el contexto de no-juego” [5], es decir, aplicar pensamientos y mecánicas de juego en entornos más serios para inducir determinados comportamientos en las personas que interactúen, así como la mejora de la participación, la motivación y el compromiso de un usuario en la realización de una tarea determinada. Es decir, la gamificación toma esas características que hacen que los juegos sean divertidos y atractivos (e incluso que sean adictivos) y los utiliza para la mejora de la experiencia del jugador en un entorno de no-juego, como puede ser en el ámbito empresarial o en la educación [6].

Como resultado de lo anterior se han desarrollado plataformas de propósito general para mejorar la motivación de los empleados o clientes. En el contexto de organizaciones que aplican BPM, resulta de interés poder aplicar gamificación en las herramientas BPMS (Business Process Management Systems), que se encargan de dar soporte automatizado al ciclo de vida de los procesos de negocio. Con todo ello, el principal objetivo del trabajo presentado en este artículo es dar soporte a la gamificación de los procesos de negocio mediante el desarrollo de una herramienta software, en la que se permita la incorporación de aspectos de juego (mecánicas, reglas y dinámicas) con el fin de fomentar la implicación y motivación de los usuarios que participan en las tareas de dichos procesos de negocio. Para ello se pretende dar soporte a la definición de juegos sobre plataformas BPMS y a su evaluación a partir de los *logs* de ejecución de dichas plataformas de modo que en base a una serie de indicadores establecidos los usuarios puedan obtener una serie de logros que puedan ser canjeados por recompensas.

El resto del artículo se organiza del siguiente modo. En el apartado 2 se describe el estado del arte, indicando algunos antecedentes para entender las contribuciones del trabajo. Posteriormente se expone el funcionamiento y el diseño de la herramienta BPMS-Game. En el apartado 4 se presenta un breve caso de estudio con la herramienta desarrollada. Y por último se presentan las principales conclusiones y las propuestas de trabajo futuro.

## 2 Estado del Arte

El campo de la gamificación ha atraído una atención significativa en varios dominios de aplicación, tales como las aplicaciones para dispositivos móviles o la educación, entre otros [7, 8]. Diversos autores han investigado sobre los beneficios de la gamificación a la hora de motivar y mejorar la implicación de los usuarios en el desarrollo de sus tareas empresariales. Diversos autores han mostrado las ventajas y beneficios de la aplicación de la gamificación, tales como Hamari et al [9] a través de una revisión de la literatura. Por su parte, la gamificación en aplicaciones webs se analizó en la revisión bibliográfica de Xu [10], concluyendo que la gamificación se ha basado tradicionalmente en las mecánicas básicas de juego (puntos, niveles, clasificación e insignias) y que deben considerarse otros aspectos más avanzados, como pueden ser características tales como la interacción social o la movilidad. Del mismo modo, en los últimos años se ha trabajado en la introducción de la gamificación en entornos de Ingeniería del Software. En [6], se presenta un mapeo sistemático de los estudios existentes en este campo, como resultado del cual se concluye que los primeros estudios primarios se remontan sólo al año 2010, por lo que es una línea de investigación bastante joven. La mayoría de los estudios se centraron en la aplicación de gamificación en la implementación del software, seguido por requisitos, gestión de proyectos y gestión de configuración.

La incorporación de mecanismos de juego en herramientas de apoyo al desarrollo de software también se ha convertido en una tendencia actual [11]. Los principales elementos empleados para la gamificación en este tipo de herramientas son: mecánicas de juego aplicadas en un ciclo desafío-recompensa-castigo, insignias o medallas, que se obtienen como resultados de los logros alcanzados, y la retroalimentación de los resultados comparando con otras personas o equipos. Existen también herramientas comerciales que dan apoyo a los procesos de ingeniería del software mediante la incorporación de los mecanismos básicos de gamificación mencionados; algunos ejemplos son Jira Hero, RedCritic, PropsToYou, ScrumKnowsy, Masterbranch o CodeHunt, entre otros. También existen algunas plataformas de gamificación que aplicadas con las herramientas corporativas de una organización ayudan a crear un entorno gamificado; alguna de estas plataformas de gamificación general son: Badgeville (<https://badgeville.com>); Gamify - (<http://www.gamify.es>); Bunchball Nitro - (<http://www.bunchball.com/products/nitro>), entre otras.

## 3 Herramienta BPMS-Game

En este apartado se describen las principales características de la herramienta BPMS-Game, que ha sido desarrollada para dar soporte a la definición de juegos sobre plataformas BPMS

### 3.1 Funcionalidades Principales

Las principales funcionalidades de la herramienta son los siguientes:

- **RF1. Extracción de información de procesos de negocio independiente del sistema BPMS.** La herramienta BPMS-Game es aplicable a cualquier sistema BPMS, para lo que se obtiene la información de ejecución de los procesos de negocio a partir de *logs* de ejecución definidos en el formato estándar XES (Extensible Event Stream).
- **RF2. Gestión de Usuarios.** La aplicación da soporte a la administración de los usuarios que participan en el juego y que son los recursos humanos participantes en el proceso de negocio. Se define el perfil de un usuario utilizando la plantilla de PERSONA [12].
- **RF3. Cálculo de medidas base.** La aplicación realiza de forma automática el cálculo de las medidas base que son usadas posteriormente para la creación de reglas.
- **RF4. Gestión de reglas y logros de juego.** La herramienta debe permitir la creación de reglas de juego, así como los logros establecidos para cada una de las reglas.
- **RF5. Canjear recompensas.** Una vez logrado los objetivos, la aplicación permitirá a los usuarios canjear sus logros por recompensas o regalos.
- **RF6. Visualización de resultados.** BPMS-Game debe permitir a los usuarios y al administrador una visualización gráfica de su progreso, así como las reglas superadas y sus recompensas.

### 3.2 Arquitectura y Tecnologías Empleadas

En la Fig 1., se muestra la arquitectura general de la herramienta. Tal como se puede observar, la herramienta incorpora la gamificación en sistemas BPMS mediante el soporte necesario para la administración de las mecánicas, reglas y dinámicas a seguirse en el juego y a la evaluación del mismo en base a la información relevante que se puede extraer de los *logs* de ejecución de los sistemas BPMS. Como formato de *logs* se considera XES [13] al tratarse del formato estándar.

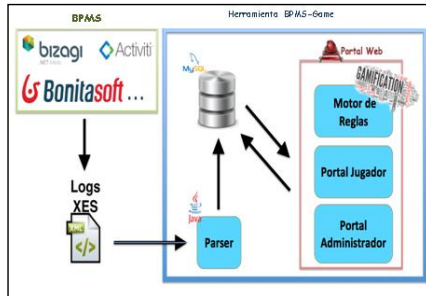


Fig. 1. BPMS-Game: Arquitectura General

Las tecnologías y herramientas empleadas para el desarrollo de la herramienta han sido: Gestión de Proyecto (Atlassian Bitbucket; Trello; Mercurial); Modelado (Visual Paradigm; Balsamic Mockups); Desarrollo (Ruby on Rails, HTML5, Xpath, JQuery). En particular destaca el uso de Ruby on Rails como framework principal para el desarrollo de la aplicación web que dota de flexibilidad y escalabilidad a BPMS-Game, así como Xpath para extraer la información de los logs.

### 3.3 Diseño de la Herramienta BPMS-Game

En este apartado se resumen los aspectos clave considerados en el diseño de BPMS-Game. En primer lugar, a partir de los requisitos establecidos se construyó el modelo de dominio de la aplicación, que se ilustra con un diagrama UML en la Fig2.

Como podemos observar en el modelo de dominio, la aplicación BPMS-Game está compuesta principalmente por los procesos que han sido extraídos de los logs de ejecución de un BPMS, los usuarios que participan, ya sea como administradores o como jugadores, y las reglas que son creadas por el administrador y que sirven para evaluar y gamificar los procesos anteriormente procesados. Cada vez que un jugador va cumpliendo alguna de las reglas definidas alcanza una recompensa, ya sea insignias (*badges*), niveles o puntos, que son las mecánicas empleadas para crear un entorno gamificado en la herramienta BPMS-Game. Los puntos que consiguen cada uno de los jugadores pueden ser canjeados por diferentes premios que el administrador ha introducido en el sistema.

Para extraer la información de los logs de ejecución en formato se ha empleado XPath, que es un lenguaje que nos permite a través de las expresiones creadas recorrer y procesar un documento XML, de modo que la información de interés extraída es almacenada en la base de datos para que pueda ser utilizada por la aplicación web para la aplicación de gamificación. De cada uno de los logs examinados se requiere extraer la siguiente información: información sobre el proceso; información sobre los casos

que tiene cada proceso; y para cada tarea incluida en los casos, el recurso humano que ejecuta dicha tarea (siempre que no se ejecute de manera automática), el número de tareas de cada recurso, el tiempo que tarda en completarlas y el coste que supone completar las tareas. Estas constituyen las medidas base sobre las cuales se definen los indicadores a evaluarse en las reglas del juego.

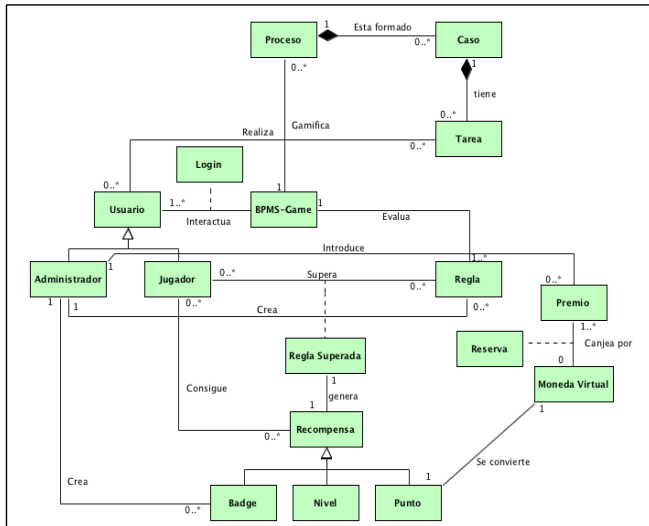


Fig. 2. Modelo de Dominio de BPMS-Game

En la Fig. 3 se muestra el formulario de definición de reglas. Toda regla tiene un identificador único y se da la opción de que la regla solo se evalúe la primera vez que un usuario la cumple, es decir, si está marcada esa opción, una vez un usuario supere dicha regla no se volverá a evaluar. De ese modo se permite definir reglas que se evalúan una sola vez o de forma repetitiva. Finalmente es obligatorio asociar a la regla una recompensa que se aplica cuando ésta sea superada. Las recompensas podrán ser puntos, niveles o insignias, que son las mecánicas de gamificación a las que la herramienta da soporte. Mediante un motor de reglas se evalúan las reglas en tiempo real a partir de la monitorización continua de los logs de ejecución.



Fig. 3. BPMS-GAME: Formulario de Definición de Reglas

Tal y como se mostraba en los requisitos de la herramienta, el Administrador debe poder crear insignias como mecánicas de la gamificación implementada. Estas insignias están formada por una imagen, un código (debe de ser único para cada una de las insignias) y un nombre representativo. Para la implementación de este caso de uso de nuevo se ha utilizado la función de *Scaffold* de Ruby on Rails.

Para la gestión de premios, se va a distinguir entre las tareas que puede realizar el rol Administrador y el rol Usuario/Jugador. Desde el punto de vista del rol Administrador, BPMS-GAME permite gestionar los diferentes premios que pueden conseguir los jugadores en función de los puntos que van consiguiendo. Para ello el Administrador debe crear premios; estos premios también podrán ser modificados o eliminados. Un atributo destacado de los premios es el Valor, este valor se define en *GCoins*, que es una moneda virtual creada para que los usuarios los puedan canjear sus puntos por los premios que el administrador ha introducido anteriormente en el sistema.

Además de todas funcionalidades ya mencionadas, también se ha añadido la opción de que los jugadores puedan compartir en las redes sociales más de moda (Facebook, Twitter y Google+) su información general del juego, como son sus puntos y el nivel en el que se encuentran. Para la implementación de esta función se ha usado los *plugins* que ofrecen las mismas redes sociales para dar soporte a dicha función.

#### 4 Aplicación de BPMS-Game: Caso de Ejemplo

Se ha realizado un caso de ejemplo, como prueba de concepto de la utilidad potencial de BPMS-Game. Para ello se considera el ejemplo básico de informe de gastos incluido en la plataforma Bonita-BPM. En la Fig. 4 se muestra el modelo BPMN2 así como un pequeño extracto del log de ejecución. Uno de los problemas encontrados al evaluar BPMS-Game es que actualmente la mayoría de herramientas BPMS representan los logs en formato propietario, por lo que fue necesario generar manualmente el log en XES a partir de la ejecución del proceso en diversos casos con varios usuarios y usando la funcionalidad que ofrece Bonita para monitorizar la ejecución de los procesos.

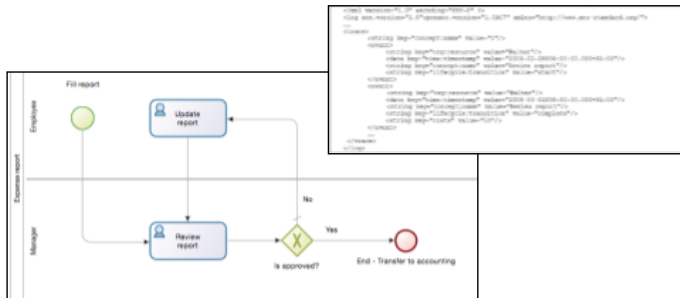


Fig. 4. Caso de Ejemplo de Aplicación de BPMS-Game

Una vez el sistema ha procesado la información de los logs XES, el administrador se encargó de crear los usuarios en la plataforma BPMS-Game con su información de perfil; las recompensas que pueden obtener los usuarios (insignias); los premios que pueden canjear; y las reglas del juego. A partir de las reglas definidas se realizó de forma automática su evaluación asignado a los usuarios la recompensa asociada a la superación de las reglas. Como ejemplo del resultado en la Fig 5., se muestra el ranking de resultados y el listado de logros obtenidos recientemente por los usuarios desde el rol del Administrador. Un usuario puede observar del mismo modo en su pantalla principal (Fig 6.): su progreso en el juego (puntos y nivel en el que se encuentra), sus logros superados, o las insignias conseguidas. Los usuarios también tienen la opción de consultar los logros que le quedan por superar o canjear los puntos conseguidos por los premios que hay en el sistema.



Fig. 5. BPMS-Game: Ranking y Logros

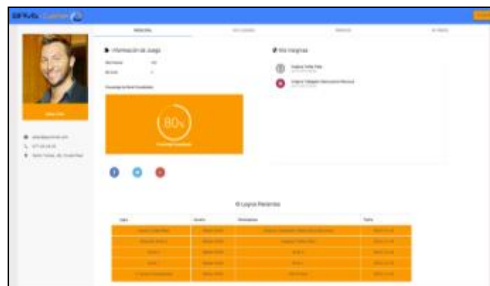


Fig. 6. BPMS-Game: Logros de un Usuario

## 5 Conclusiones y trabajo futuro

En este artículo se ha presentado BPMS-Game, herramienta que da soporte a la definición de los juegos sobre plataformas BPMS y a su evaluación a partir de los *logs* de ejecución de dichas plataformas. La aportación que ofrece BPMS-Game en el campo de BPM es que, al introducir la gamificación en este campo, permite fomentar la implicación y motivación de los usuarios que participan en las tareas de los procesos de negocio, ya que las personas son uno de los componentes principales en los que se basa el BPM y sus aplicaciones BPMS. Como trabajos futuros se pretende incluir un módulo de gestión de indicadores, ya que actualmente la herramienta da soporte una serie de indicadores básicos, de modo que permita al administrador definir nuevos indicadores en base a cálculos extraídos de la información de los *logs*, dando más flexibilidad a la herramienta. También se pretende añadir nuevas funcionalidades

sociales a la herramienta, como son una gamificación por equipos, o añadir un chat con el cual los usuarios puedan comunicarse. Este tipo de funcionalidades son muy importantes ya que ayudan a fomentar el trabajo en equipo y cooperativo. Otro reto para alcanzar en un futuro es poner en marcha el sistema en empresas, para así poder realizar casos de estudios reales.

### Agradecimientos

Este trabajo ha sido parcialmente financiado por los siguientes proyectos: GINSENG (Ministerio de Economía y Competitividad y Fondo Europeo de Desarrollo Regional FEDER, TIN2015-70259-C2-1-R), proyecto INGENIOSO (PEII-2014-050-P), Junta de Comunidades de Castilla-La Mancha y FEDER) y Red en Ciencia e Ingeniería de Servicios (TIN2014-53986-REDT).

### Referencias

1. Weske, M., *Business Process Management: Concepts, Languages, Architectures*. Springer Publishing Company, (2010).
2. Dumas, M., et al., *Fundamentals of Business Process Management*. Springer Publishing Company, Incorporated. (2013).
3. Brocke, J.v. and M. Rosemann, *Handbook on Business Process Management I: Introduction, Methods, and Information Systems*. Springer Publishing Company. (2010).
4. Trkman, P., *The critical success factors of business process management*. Int. J. Inf. Manag. **30**(2): p. 125-134 (2010).
5. Deterding, S., et al., *From game design elements to gamefulness: defining "gamification"*, in *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*. ACM: Tampere, Finland. p. 9-15 (2011).
6. Pedreira, O., et al., *Gamification in software engineering – A systematic mapping*. Information and Software Technology. **57**: p. 157-168 (2014).
7. Hugos, M., *Enterprise Games: Using Game Mechanics to Build a Better Business*. O'Reilly (2012).
8. Zichermann, G. and C. Cunningham, *Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps*. O'Reilly Media, Inc. 208. (2011)
9. Hamari, J., J. Koivisto, and H. Sarsa, *Does Gamification Work? – A Literature Review of Empirical Studies on Gamification*, in *Proceedings of the 2014 47th Hawaii International Conference on System Sciences*. IEEE Computer Society. p. 3025-3034. (2014)
10. Xu, Y., *Literature Review on Web Application Gamification and Analytics*. Department of Information and Computer Sciences, University of Hawaii: Collaborative Software Development Lab (CSDL). (2012)
11. Passos, E.B., et al., *Turning Real-World Software Development into a Game*, in *Proceedings of the 2011 Brazilian Symposium on Games and Digital Entertainment*. IEEE Computer Society. p. 260 - 269. (2011)
12. Kumar, J. and M. Heger, *Gamification at Work: Designing Engaging Business Software*. Interaction Design Foundation. (2013).
13. Eindhoven, T.U., *XES. Standard Definition*. 2014.

# Análisis inteligente de flujos de trabajo sociales \*

Manuel Lama<sup>1</sup>, Pedro Álvarez<sup>2</sup>, Manuel Ocaña<sup>3</sup>, Manuel Mucientes<sup>1</sup>, Joaquín Ezpeleta<sup>2</sup>, Miguel Ángel Garrido<sup>3</sup>, Alberto Bugarín<sup>1</sup>

<sup>1</sup> Centro Singular de Investigación en Tecnoloxías da Información (CiTIUS)  
Universidade de Santiago de Compostela, Santiago de Compostela, España  
{manuel.lama, manuel.mucientes, alberto.bugarin.diz}@usc.es

<sup>2</sup> Instituto de Investigación en Ingeniería de Aragón (I3A)  
Depto. de Informática e Ingeniería de Sistemas, Universidad de Zaragoza, España  
{alvaper, ezpeleta}@unizar.es

<sup>3</sup> Depto. de Electrónica, Escuela Politécnica Superior.  
Universidad de Alcalá de Henares, Alcalá de Henares, España.  
{mocana, garrido}@depeca.uah.es

**Abstract.** Los flujos de trabajo sociales (SOW) coordinan las actividades realizadas por un conjunto de usuarios que bien de forma individual o en cooperación tratan de alcanzar un determinado objetivo. Los SOW son flujos no estructurados en los que participan un gran número de usuarios que llevan a cabo actividades de muy diversa naturaleza que se extienden a lo largo del tiempo y que típicamente consumen pocos recursos de computación. Un ejemplo de procesos que se modelan a través de este tipo de flujos de trabajo son las campañas de marketing que tienen como objetivo motivar a los potenciales clientes en el consumo de un determinado producto o servicio. En este trabajo, se presenta el proyecto *Inteligencia (Artificial) de Negocio para Flujos de Trabajo Sociales*, en el que se combinan técnicas de minería de procesos, estrategias de paralelización de algoritmos, y técnicas de localización y seguimiento de usuarios, con el fin de extraer información relevante sobre SOW, como los que modelan, entre otros, el comportamiento de los usuarios en campañas de marketing desarrolladas en escenarios abiertos.

## 1 Introducción

Los *flujos de trabajo sociales* (SOW) modelan la coordinación de la ejecución de una serie de actividades que deben ser realizadas por un conjunto de usuarios [7]. Estos usuarios pueden actuar de forma individual o colaborativa para alcanzar un determinado objetivo, potenciando así el carácter social del flujo de trabajo. Ejemplos de este tipo de flujos de trabajo (WF) son las campañas de marketing que se realizan en espacios abiertos o la formación de empleados a través de estrategias de *gamificación*. No obstante, desde un punto de vista tecnológico las

\* Los autores desean agradecer al Ministerio de Economía y Competitividad el soporte financiero a través de los proyectos TIN2014-56633-C3-1-R, TIN2014-56633-C3-2-R, TIN2014-56633-C3-3-R y TIN2014-53986-REDT.

principales características de los SOW son: (i) modelan *procesos no estructurados* en los que los usuarios disponen de múltiples opciones y/o caminos y donde incluso pueden realizar actividades adicionales a las inicialmente previstas en el flujo de trabajo; (ii) pueden participar simultáneamente un elevado número de usuarios; y (iii) en función de la habilidad o motivación de cada usuario, una misma actividad puede ser realizada de inmediato o tener una duración temporal mucho mayor (del orden de horas o incluso días).

Estas características añaden una serie de retos a la hora de gestionar y monitorizar la ejecución de los SOW. Por una parte, la gestión de este tipo de WF requiere el manejo de un gran número de ejecuciones simultáneas cuya duración se extiende a lo largo del tiempo. Por otra parte, el análisis de la ejecución de los WF requiere del uso de técnicas de minería de procesos [1] para obtener el WF *real* seguido por los usuarios y, de este modo, *entender* qué es lo que realmente ha pasado en la ejecución del proceso, permitiendo con ello la mejora del proceso y su adaptación (dinámica) a las necesidades de los usuarios. Sin embargo, para procesos no estructurados con un número enorme de ejecuciones, los resultados de los algoritmos actuales de descubrimiento de procesos son WF que tienen una estructura en *espagueti* y que, por tanto, son muy difíciles, cuando no imposibles, de interpretar. Por ello, es necesario desarrollar técnicas que permitan extraer información valiosa sobre el flujo de trabajo descubierto. Este es el primer eje alrededor del cual pivota el proyecto *Inteligencia (Artificial) de Negocio para Flujos de Trabajo Sociales* (BAI4SOW).

Por otra parte, el proyecto BAI4SOW añade otra dimensión a los SOW: las actividades a realizar por parte de los usuarios tienen lugar en escenarios o áreas geográficas de exteriores (como puede ser un centro comercial abierto, una zona de interés turístico, un área donde se desarrolle alguna actividad o, en general, cualquier zona de una ciudad) y además son actividades geoposicionadas que los usuarios llevan a cabo a través de dispositivos móviles (como las actividades de una campaña de marketing). Por tanto, es necesario capturar los eventos relacionados con la ejecución de este tipo de actividades y, para ello, se requiere el desarrollo de técnicas de localización y de detección del comportamiento de los usuarios en un área geográfica dada. Este es el segundo eje del proyecto BAI4SOW.

En este artículo, se presenta el proyecto BAI4SOW cuyo objetivo consiste en el desarrollo de algoritmos de minería de procesos para el análisis de SOW que contienen actividades geoposicionadas. Además, dado el elevado número de usuarios que potencialmente pueden participar en este tipo de flujos es necesario ejecutar los algoritmos en infraestructuras de cómputo *grid*, *clúster* y *cloud*, minimizando el coste que podría tener dicho cómputo en *clouds* empresariales como Amazon y Google. Este es el tercer eje del proyecto BAI4SOW.

El artículo se estructura de la siguiente manera: en la Sección 2 se presenta el marco conceptual del proyecto; en las Secciones 3, 4 y 5 se describe cómo se localiza a los usuarios, los algoritmos de minería de procesos y la capa *middleware* que gestiona las infraestructuras de cómputo, respectivamente. Finalmente, la Sección 6 presenta las conclusiones del trabajo y el trabajo futuro.

## 2 Marco conceptual

BAI4SOW tiene como objetivo general el desarrollo de técnicas inteligentes para la extracción automática y el análisis del comportamiento de los usuarios en procesos modelados a través de flujos de trabajo sociales, en general, y en particular, en aquellos procesos que involucren a usuarios que se mueven en un área geográfica exterior y que ejecutan las actividades del flujo social a través de sus dispositivos móviles. Para alcanzar este objetivo se ha propuesto un marco conceptual que consta de los siguientes componentes (Figura 1):

- *Algoritmos de localización y detección de comportamientos.* Para analizar los SOW es necesario registrar el conjunto de eventos generados por las actividades que realiza cada usuario en una ejecución de dichos flujos, de modo que para cada ejecución se registra una traza de estos eventos. En procesos que involucren el desplazamiento de los usuarios en un área geográfica, estas actividades consisten en llegar a un punto determinado (para, por ejemplo, realizar una compra o tomar una foto), sentarse o moverse dentro de un local, etc. Por tanto, es necesario *localizar* a los usuarios en el área geográfica sobre la que se mueven, así como detectar sus comportamientos de interés una vez se encuentran en una localización dada. Para registrar la realización de estas actividades se desarrollan una serie de algoritmos que hacen uso del GPS, de la señal WiFi y de los sensores inerciales de los dispositivos móviles de los usuarios.
- *Algoritmos de minería de procesos.* Este conjunto de algoritmos permiten, por una parte, *descubrir automáticamente* el WF *real* que ha sido ejecutado por los usuarios y, por otra parte, extraer información relevante sobre dicho WF, como los *patrones frecuentes* de actividades, que indican modelos

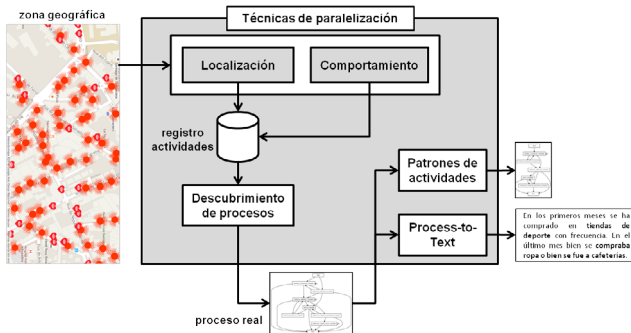


Fig. 1. Marco conceptual del proyecto.

de comportamiento frecuente por parte de los usuarios; la *jerarquización* de WF no estructurados y en *espagueti*, que ayuda a visualizar el proceso al reducir su complejidad en diferentes niveles de abstracción; y la *descripción lingüística* de dichos WF, con la que se relatan de forma automática y en lenguaje natural sus características más relevantes, proporcionando una información complementaria a la que obtienen los usuarios mediante la visualización a través de grafos.

- *Técnicas de paralelización de algoritmos*. Se hará uso de este tipo de técnicas para diseñar una infraestructura tecnológica basada en la integración de recursos *grid* y *cloud* que ofrezca soporte para el procesamiento eficiente y a gran escala de los algoritmos desarrollados en el marco del proyecto. Esta infraestructura integrará modelos de coste que faciliten la decisión de qué recursos computacionales y de almacenamiento son más apropiados en cada instante para la ejecución de los algoritmos, contemplando la posibilidad de utilizar recursos de diferentes infraestructuras o proveedores.

Es importante resaltar que este marco conceptual es aplicable a SOW que contienen actividades relacionadas con la localización y el comportamiento de los usuarios que participan en ellos. No obstante, las técnicas de minería de procesos y de paralelización de algoritmos son directamente aplicables a cualquier WF, en general, y en particular a flujos no estructurados y en los que participan un enorme número de usuarios. En las siguientes secciones se detallan cada uno de los componentes del marco conceptual.

### 3 Localización y detección de comportamientos

El objetivo de este componente del marco conceptual es doble: por una parte, desarrollar un *sistema de localización* de usuarios en zonas de interés tanto para interiores como exteriores, y por otra parte, identificar y reconocer de forma automática el conjunto de actividades que definen el comportamiento de un usuario cuando se encuentra en un punto geográfico dado (por ejemplo, un local comercial). Así:

- El proceso de localización está basado en un sistema jerárquico que proporciona la estimación de la posición del usuario en una zona geográfica dada (como un centro comercial), empleando para ello: (i) un sistema de localización de alto nivel que identifica las zonas o puntos de interés por medio del empleo del GPS en exteriores y el sistema de localización con *fingerprints* propuesto por los autores en [9] enriquecido con (ii) un sistema de localización de bajo nivel que realiza el seguimiento entre posiciones o zonas de interés, haciendo una fusión multisensorial basada en Filtro Extendido de Kalman (EKF) con los sensores inerciales del dispositivo móvil (giróscopos, acelerómetros y podómetros), como se muestra en la Figura 2. Para mejorar el cálculo de la orientación se suele recurrir al uso de filtros, como el filtro de Mahony [11], que es un filtro complementario eficiente para sensores inerciales, aunque tiene un error constante en la orientación, al no



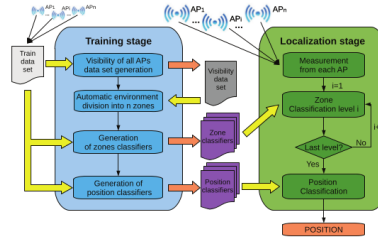


Fig. 2. Localizador de alto nivel.

disponer de un observador absoluto ortogonal a la gravedad. Para resolver este problema se puede usar el filtro de Madgwick [10], como un filtro de gradiente descendiente optimizado que calcula la orientación incluyendo la compensación de la distorsión magnética y la compensación del bias del giróscopo. De esta manera, añade al filtro de Mahony las medidas del magnetómetro y, por tanto, se obtiene la orientación en las tres dimensiones. Así, para obtener la orientación de la unidad de medida inercial (IMU) del dispositivo móvil (Figura 3), en primer lugar se calibrará y, a continuación, se usará el filtro de Madgwick para compensar el *bias* del giróscopo. Posteriormente, se fusionan estos datos con el podómetro del móvil mediante el filtro EKF.

- La detección del comportamiento de los usuarios exige describir cualitativamente el conjunto de comportamientos (o actividades) que se desea detectar. Para ello se usa un modelo lingüístico jerárquico con varios niveles de abstracción que describe de forma incremental las actividades de los usuarios,

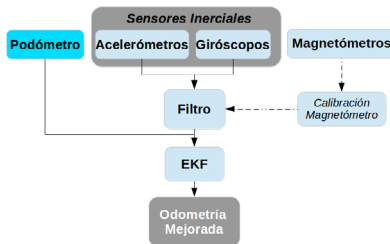


Fig. 3. Sistema de seguimiento propuesto.

estableciendo hipótesis a partir de la información de eventos de más bajo nivel proporcionados por los sensores disponibles en el dispositivo móvil. El paso de datos sensoriales de bajo nivel a información cualitativa de alto nivel se realizará mediante un algoritmo inteligente de fusión multisensorial basado en técnicas de *soft computing* adaptadas a entornos de procesamiento masivo que permiten manejar los datos de miles de usuarios simultáneamente.

Como resultado, este componente genera y registra los eventos asociados con la ejecución de las actividades de localización y de comportamiento de usuarios que forman parte de los SOW. Estos eventos contienen información sobre cuándo y dónde ha tenido lugar la actividad, así como quién la ha realizado. Esta información será la que usarán los algoritmos de minería de procesos para obtener los WF *reales* que han sido seguido por los usuarios.

#### 4 Algoritmos de minería de procesos

Las técnicas de minería de procesos [1] permiten descubrir automáticamente, y a partir de ficheros de registro, el WF *real*, o comportamiento, que han seguido los usuarios que participan en un SOW. Estos ficheros de registro presentan dos características que condicionan enormemente el tipo de algoritmos a desarrollar: la gran cantidad de usuarios que potencialmente pueden participar en un SOW (del orden de decenas de miles) y el elevado número de eventos asociados a actividades que puede generar cada usuario (decenas de actividades). En estos casos, los WF descubiertos por los algoritmos tienen una estructura en *espagueti* que dificulta su comprensibilidad, ya que existe un enorme número de relaciones causales entre las actividades que conforman el WF. Para facilitar esta comprensibilidad por parte de los gestores del WF, se aplicarán las siguientes técnicas de minería de procesos:

- *Algoritmo de descubrimiento de WF*. Este algoritmo debe generar WF que sean completos, precisos y simples, considerando los elevados niveles de ruido que contienen los registros de eventos de actividades en los SOW. Así, el algoritmo debe maximizar los criterios de completitud, precisión y simplicidad, ya que pueden existir caminos de control poco frecuentes pero que son muy informativos, filtrando en la medida de lo posible el ruido que añaden los usuarios al completar actividades no informativas o no completar los WF. Uno de los algoritmos que tiene en cuenta estos requisitos es ProDiGen [13], un algoritmo evolutivo basado en una función jerárquica de evaluación que prioriza la completitud frente a la precisión y ésta frente a la simplicidad. Este algoritmo está integrado en la plataforma ProDiGen que se está desarrollando en el marco del proyecto (Figura 4).
- *Algoritmo de identificación de patrones frecuentes*. A través de este algoritmo, y partiendo del fichero de registro y del WF descubierto, se identifican los patrones de actividades que los usuarios realizan con *frecuencia* durante la ejecución del WF. De esta forma, se describen únicamente aquellos patrones que son ejecutados con una frecuencia superior a un umbral dado,

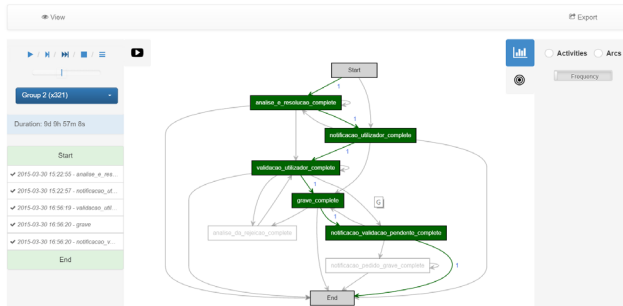


Fig. 4. Herramienta ProDiGen de minería de procesos.

lo que permite entender el comportamiento de los usuarios. Para ello se ha desarrollado un algoritmo basado en el algoritmo *w-find* [8] que ha sido extendido para dar soporte a la identificación de patrones de actividades con bucles de cualquier longitud [4]. Este algoritmo también está integrado en la plataforma ProDiGen.

- *Algoritmo de jerarquización de WF*. El objetivo de este algoritmo consiste en proporcionar diferentes niveles de abstracción de los WF descubiertos [3], obteniendo representaciones más simples en cada uno de estos niveles y facilitando con ello la legibilidad del WF. El algoritmo de jerarquización hace uso de técnicas de agrupamiento de trazas en las que los grupos de trazas se crean en función de su similitud.

Además de este conjunto de algoritmos basados en técnicas de minería de procesos con los que se pretende describir visualmente el comportamiento de los usuarios, se desarrollará una herramienta [12] orientada a la *descripción lingüística* de los WF reales que han sido obtenidos por el algoritmo de descubrimiento. Esta herramienta actuará como un generador automático de informes en lenguaje natural a partir del proceso descubierto (herramienta *process-to-text*), proporcionando información relevante y seleccionada sobre aquellos aspectos de interés del WF, tales como los patrones y caminos frecuentes, las actividades realizadas un mayor número de veces en un período dado, o la frecuencia de las relaciones entre actividades. Es importante resaltar que estas descripciones textuales son complementarias a la representación visual de los WF a través de grafos de actividades.

## 5 Paralelización de algoritmos

En el proyecto BAI4SOW, los SOW modelan procesos en los que potencialmente pueden participar miles o decenas de miles de usuarios, como puede ser las campañas de marketing en centros comerciales o los procesos de formación en grandes compañías. Esta característica conlleva la necesidad de paralelizar los algoritmos desarrollados y de ejecutar estas versiones paralelas en infraestructuras de cómputo *grid*, *clúster* y *cloud*.

El punto de partida es una capa software (a nivel de *middleware*) capaz de integrar una serie de infraestructuras de cómputo *clúster* y *grid* que tenemos a nuestra disposición [5]. Esta capa ofrece la funcionalidad típica de un *framework de gestión de recursos*. Todos los recursos de estas infraestructuras son vistos como un único y potente recurso de cómputo y almacenamiento sobre el que es posible ejecutar las versiones paralelas de los algoritmos de minería de procesos y de localización y detección del comportamiento de los usuarios. La Figura 5 muestra el diseño arquitectural de esta capa intermediaria. Su arquitectura está basada en un bus de mensajes que coordina la actividad de los componentes que soportan el ciclo de vida de la ejecución de una aplicación intensa en cómputo y datos. Entre los componentes existentes están los responsables de la gestión y provisión de los recursos, el *scheduling* de las tareas a ejecutar, la gestión y recuperación de fallos, el movimiento de datos entre recursos, la

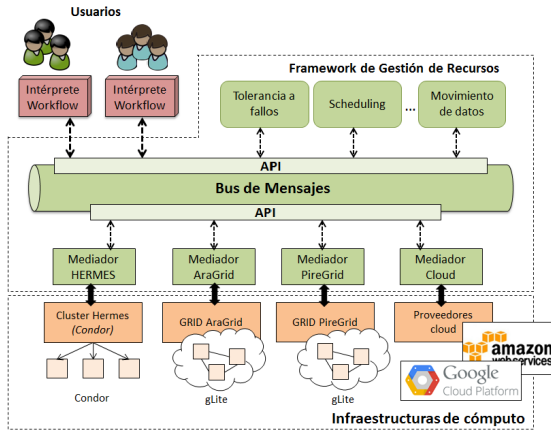


Fig. 5. Arquitectura de alto nivel de la capa *middleware*

monitorización del estado de ejecución, y los mediadores que facilitan el acceso a las infraestructuras de cómputo que fueron integradas. Esta capa software ha sido anteriormente utilizada para resolver con éxito problemas que presentaban fuertes requisitos de cómputo y datos, por ejemplo, la implementación paralela del sistema de anotación semántica presentado en [6].

En BAI4SOW se pretenden potenciar las capacidades de la capa *middleware* para su uso y explotación en entornos de empresa. En la mayoría de los casos, las infraestructuras de cómputo disponibles en el contexto académico (*clúster* o *grid*) no están pensadas para dar soporte a la ejecución de sistemas en producción: sus recursos sólo pueden ser usados para fines relacionados con la investigación, o su fiabilidad, disponibilidad y rendimiento no son adecuados. Para resolver este problema, la funcionalidad de la capa *middleware* será extendida para integrar y gestionar eficientemente los recursos y servicios ofrecidos por los actuales proveedores de *cloud* a nivel de *Infrastructure-as-a-Service* (Amazon AWS, Google Cloud Platform o Microsoft Azure, entre otros). Un primer prototipo de esta nueva versión del *middleware* ya ha sido implementado y probado en el entorno de Amazon AWS. Este prototipo a día de hoy permite ejecutar sobre recursos *cloud* los distintos algoritmos de minería de procesos y de localización y detección del comportamiento de los usuarios, abriendo de esta manera la posibilidad de explotar nuestros resultados en entornos reales de empresa. No obstante, esta evolución hacia entornos tipo *cloud* ha abierto una nueva línea de trabajo en la que abordan cuestiones críticas como, por ejemplo, ¿cómo minimizar el coste de ejecutar estas aplicaciones en una infraestructura donde se paga por el uso de los recursos?, ¿qué recursos son los más adecuados para ejecutar una aplicación o parte de una aplicación?, ¿dónde y cómo almacenar los datos teniendo en cuenta los recursos que han sido contratados?, etc. Todas estas cuestiones se enmarcan en lo que se ha denominado la *economía del cloud*.

En [2] definimos un primer intento de responder a las cuestiones anteriores para el caso de ejecutar aplicaciones del tipo *bag-of-tasks* sobre Amazon AWS. Un modelo de minimización de costes aplicado sobre el catálogo de recursos del proveedor pretendía determinar la mejor combinación de recursos para ejecutar una aplicación teniendo en cuenta los requisitos del usuario (presupuesto o tiempo máximo de ejecución, por ejemplo) y el rendimiento de las instancias contratadas. En BAI4SOW será necesario definir nuevos modelos de minimización que se adecuen a los algoritmos involucrados e integrar estos modelos en el ciclo de vida de la capa *middleware* previamente desarrollada. El objetivo final debe ser que el aprovisionamiento de los recursos *cloud* sea bajo criterios de coste y rendimiento, y el *scheduling* de las aplicaciones sobre estos recursos sea automático desde la perspectiva del usuario final.

## 6 Conclusiones

En el proyecto BAI4SOW se combinan de forma exitosa técnicas de minería de procesos, estrategias de paralelización de algoritmos, y técnicas de localización y seguimiento de usuarios para extraer información valiosa sobre la ejecución de

SOW en los que los usuarios realizan actividades en escenarios abiertos como centros comerciales abiertos.

Actualmente, el proyecto se encuentra en una fase de paralelización de los algoritmos que ya han sido desarrollados para que puedan ser desplegados en las infraestructuras de cómputo *grid* y *clúster*. Una vez se haya completado ese despliegue se realizarán una serie de pilotos en entornos con usuarios reales.

## References

1. van der Aalst, W.M.P.: Process Mining - Discovery, Conformance and Enhancement of Business Processes. Springer (2011)
2. Álvarez, P., Hernández, S., Fabra, J., Ezepeleta, J.: Cost estimation for the provisioning of computing resources to execute bag-of-tasks applications in the amazon cloud. In: 12th International Conference on Economics of Grids, Clouds, Systems and Services (GECON) 2015. Cluj-Napoca, Romania (2015)
3. Bose, R.P.J.C., Verbeek, H.M.W.E., van der Aalst, W.M.P.: Discovering hierarchical process models using prom. In: Nurcan, S. (ed.) Proceedings of the CAiSE Forum 2011. CEUR Workshop Proceedings, vol. 734, pp. 33–40. London, UK (2011)
4. Chapela, D., Vazquez-Barreiros, B., Mucientes, M., Lama, M.: Mining frequent pattern in workflows. In: International Workshop on Algorithms & Theories for the Analysis of Event Data (ATAED 2016) (2016), (Submitted)
5. Fabra, J., Álvarez, P., Bañares, J., Ezepeleta, J.: DENEb: a platform for the development and execution of interoperable dynamic web processes. Concurrency and Computation: Practice and Experience 23(18), 2421–2451 (2011)
6. Fabra, J., Hernández, S., Otero, E., Vidal, J.C., Lama, M., Álvarez, P.: Integration of grid, cluster and cloud resources to semantically annotate a large-sized repository of learning objects. Concurrency and Computation: Practice and Experience 27(17), 4603–4629 (2015)
7. Gorg, S., Bergmann, R.: Social workflows - Vision and potential study. Information Systems 50, 1–19 (2015)
8. Greco, G., Guzzo, A., Manco, G., Saccà, D.: Mining and reasoning on workflows. IEEE Transactions on Knowledge & Data Engineering 17(4), 519–534 (2005)
9. Hernández, N., Alonso, J.M., Ocaña, M.: Hierarchical approach to enhancing topology-based WiFi indoor localization in large environments. Journal of Multiple-Valued Logic and Soft Computing 3(5), 221–241 (2016)
10. Madgwick, S.O.: An efficient orientation filter for inertial and inertial/magnetic sensor arrays. Report x-io and University of Bristol (UK) (2010)
11. Mahony, R., Hamel, T., Pfimlin, J.M.: Nonlinear complementary filters on the special orthogonal group. IEEE Transactions on Automatic Control 53(5), 1203–1218 (2008)
12. Ramos-Soto, A., Bugarín, A., Barro, S.: On the role of linguistic descriptions of data in the building of natural language generation systems. Fuzzy Sets and Systems 285, 31–51 (2016)
13. Vázquez-Barreiros, B., Mucientes, M., Lama, M.: ProDiGen: Mining complete, precise and minimal structure process models with a genetic algorithm. Information Sciences 294, 315–333 (2015)

# RALph: A Graphical Notation for Resource Assignments in Business Processes (Abstract)\*

Cristina Cabanillas<sup>1</sup>, David Knuplesch<sup>2</sup>, Manuel Resinas<sup>3</sup>,  
Manfred Reichert<sup>2</sup>, Jan Mendling<sup>1</sup>, and Antonio Ruiz-Cortés<sup>3</sup>

<sup>1</sup>Vienna University of Economics and Business, Austria  
{cristina.cabanillas, jan.mendling}@wu.ac.at

<sup>2</sup>Ulm University, Germany {david.knuplesch, manfred.reichert}@uni-ulm.de

<sup>3</sup>University of Seville, Spain {resinas, aruiz}@us.es

## Summary of the Contribution

The organisational perspective of business processes deals with the management of human resources throughout the process lifecycle. This involves the definition of assignments at design time, i.e. by querying those actors that are allowed to work on tasks; the allocation of resources at runtime; and the analysis of resource utilisation after execution for process improvement. While it is widely accepted that models and visual notations can be beneficial for system development [1], it is striking to note that a notation for modelling these aspects in an integrated way is still missing. Although a few works have recently contributed towards a better integration of a visual notation for defining resource assignments with extensive semantics [2, 3], they still expose gaps towards a full visual support.

We aim at bridging this gap by introducing RALph, a graphical notation for defining the assignments of human resources to process activities. RALph has the following characteristics: (i) It is expressive. In particular, it allows defining all the resource selection conditions covered by the workflow resource patterns [4] as well as those we discovered in a real scenario from the healthcare domain. (ii) Resource assignments specified with RALph can be automatically analysed. In turn, this enables automatic answers to questions such as “Is the process consistent regarding the use of resources?” or “Which activities may Mr. B perform in the context of process X?”. This is achieved by defining the semantics of RALph through a semantic mapping to Resource Assignment Language (RAL) [2], a textual language for resource assignment whose formal semantics is defined in description logics. (iii) It is independent of any business process modelling language. For that, it can be seamlessly integrated with existing notations (e.g., Business Process Model and Notation (BPMN)), as demonstrated with a proof-of-concept prototype that we have developed.

---

\* Cristina Cabanillas, David Knuplesch, Manuel Resinas, Manfred Reichert, Jan Mendling, Antonio Ruiz-Cortés: RALph: A Graphical Notation for Resource Assignments in Business Processes. International Conference on Advanced Information Systems Engineering (CAiSE) 2015: 53-68. DOI: 10.1007/978-3-319-19069-3\_4

## References

1. J. Whittle, J. Hutchinson, and M. Rouncefield, "The State of Practice in Model-Driven Engineering," *IEEE Software*, vol. 31, no. 3, pp. 79–85, 2014.
2. C. Cabanillas, M. Resinas, A. del Rfo-Ortega, and A. Ruiz-Cortés, "Specification and Automated Design-Time Analysis of the Business Process Human Resource Perspective," *Inf. Syst.*, vol. 52, pp. 55–82, 2015.
3. M. Strembeck and J. Mendling, "Modeling process-related RBAC models with extended UML activity models," *Inf. Softw. Technol.*, vol. 53, pp. 456–483, 2011.
4. N. Russell, W. M. P. van der Aalst, A. H. M. ter Hofstede, and D. Edmond, "Workflow Resource Patterns: Identification, Representation and Tool Support," in *CAiSE*, pp. 216–232, 2005.





V Congreso Español de Informática  
Salamanca, 13 al 16 de septiembre de 2016



Organizadores:



SCIE  
SOCIEDAD  
CIENTÍFICA  
INFORMÁTICA  
DE ESPAÑA



UNIVERSIDAD  
DE SALAMANCA  
CAMPUSES EXCELENTEN EN INTERNACIONAL



GRUPO DE INVESTIGACIÓN  
BISITE

ISBN: 978-84-9012-628-8 (pdf)