

## Business process automatic verification with a compositional approach

Luis E. Mendoza Morales<sup>1</sup>, Manuel I. Capel Tuñón<sup>2</sup>, María A. Pérez<sup>1</sup>

<sup>1</sup>Departamento de Procesos y Sistemas, Universidad Simón Bolívar, Apartado Postal 89000,  
Caracas 1080-A, Venezuela. {lmendoza, movalles}@usb.ve

<sup>2</sup>Departamento de Lenguajes y Sistemas Informáticos, Universidad de Granada,  
ETSI de Informática y Telecomunicación. 18071 Granada, España. manuelcapel@ugr.es

### Abstract

Currently, complex *Business Processes* (BP) are implemented by *Enterprise Information Systems* (EIS). However, the development of EIS is complicated if there is no guarantee the reliability of the *Task Model* (TM) of the PN. This article describes the *Formal Compositional Verification Approach* (FCVA), an infrastructure for the formal specification and verification of BP, supported by *Model Checking* (MC) techniques. Based on a proposal for a temporal formal semantics for the *Business Process Model and Notation* (BPMN), which allows us to obtain the MT of a business process, the EFVC supports MT behavior verification from the individual verification of their participants. The interpretation of notational elements of BPMN in the *Communicating Sequential Processes + Time* (CSP+T) formal language allows us to incorporate MC tools widely used to verify the behaviour of a TM. A real-life example in the field of the *Customer Relationship Management* (CRM) business is presented to demonstrate the FCVA application in a practical way.

**Key words:** business processes, tasks model, model checking, compositional verification, formal specification.

## Verificación automática de procesos de negocio con un enfoque composicional

### Resumen

En la actualidad, los *Procesos de Negocio* (PN) complejos son implementados por *Sistemas de Información Empresariales* (SIE). Sin embargo, el desarrollo de los SIE se ve dificultado sino se garantiza la confiabilidad del *Modelo de Tareas* (MT) de los PN. Este artículo describe el *Enfoque Formal de Verificación Composicional* (EFVC); una infraestructura para la especificación formal y verificación de PN, soportado por técnicas de *Verificación Automática* (VA). Basado en una propuesta de semántica formal temporal para la *Business Process Model and Notation* (BPMN) que permite obtener el MT de un proceso de negocio, el EFVC soporta la verificación del comportamiento del MT a partir de la verificación individual de sus participantes. La interpretación de los elementos notacionales de BPMN en el lenguaje formal *Communicating Sequential Processes + Time* (CSP+T) nos permite incorporar herramientas de VA ampliamente utilizadas para verificar el comportamiento de un MT. Se presenta un ejemplo de aplicación real en el área del *Manejo de las Relaciones con el Cliente* (conocido como CRM) para demostrar el uso práctico de EFVC.

**Palabras clave:** procesos de negocio, modelo de tareas, verificación automática, verificación composicional, especificación formal.

## Introducción

En muchas empresas, los *Sistemas de Información Empresariales* (SIE) implementan la ejecución de *Procesos de Negocio* (PN) cross-funcionales complejos, los cuales permiten alcanzar los objetivos del negocio y los requisitos de los clientes, manteniendo la comunicación con socios del negocio, trascendiendo los objetivos de límites de ventas, marketing, manufactura e investigación y desarrollo. Por tanto, una organización que busque orientar su gestión sobre la base de sus PN, debe obtener, como resultado de un *Modelado de Procesos de Negocio* (MPN), una definición completa, consistente, confiable y segura del *Modelo de Tareas* (MT) de los PN que implementen sus SIE y describa la realización de las actividades a que dan lugar los PN. La verificación de un MT no debería posponerse hasta la fase de pruebas de los SIE, después de su implementación, dado que proceder así aumenta el costo y el tiempo de entrega de los sistemas final. Una alternativa cada vez más adoptada consiste en iniciar la verificación en la fase de análisis.

La falta de una semántica precisa de las entidades de modelado de la *Business Process Model and Notation* (BPMN) [1], impide llevar a cabo un análisis y razonamiento rigurosos de los modelos que pueden obtenerse con BPMN [2], como parte de un MPN. Para hacer frente a esta situación, se propone una adaptación de nuestro *Enfoque Formal de Verificación Composicional* (EFVC) [3], fundamentado en técnicas de *Verificación Automática* (VA). Esta adaptación permite verificar un MT con la ayuda de una semántica temporal que proponemos, basada en el lenguaje de especificación formal *Communicating Sequential Processes + Time* (CSP+T) [4], para el subconjunto de entidades de modelado de BPMN que permiten especificar actividades y eventos del negocio que dependen del tiempo. La propuesta utiliza el lenguaje de especificación CSP+T, el cual tiene una forma simple pero potente de composición de sus términos de procesos sintácticos dada por los operadores de composición concurrente y de ocultamiento. Utilizando toda esta infraestructura formal, el EFVC hace factible la verificación composicional del comportamiento global de los PN (el MT) a partir del comportamiento de los participantes que colaboran en la realización del MT.

En la literatura se puede encontrar diferentes trabajos que abordan la verificación de PN modelados con BPMN. Existen métodos formales de verificación de modelos BPMN basados en el  $\pi$ -cálculo [5] o en Redes de Petri [6], entre otros. Morimoto [7] presenta un amplio estudio de las técnicas de verificación propuestas recientemente, así como una comparación entre ellas con respecto a sus motivaciones, métodos y lógicas. Sin embargo, ninguno de los trabajos combina el modelado de PN con la especificación, el diseño y la verificación del MT; ningún trabajo aprovecha al máximo las fortalezas de los cálculos de procesos aplicados al área de la verificación de PN. Nuestro trabajo ofrece una visión sistémica e integrada de la especificación, el diseño y las tareas de verificación de un MT derivado de los PN. La incorporación de técnicas de VA en la especificación y verificación de un MT, como parte del desarrollo de los SIE, ha permitido garantizar la confiabilidad y seguridad de ambos.

El resto de este artículo se organiza de la siguiente manera. En la sección 2 se da una breve introducción de CSP+T previo a la descripción de la semántica formal de las entidades de modelado BPMN con anotaciones temporales. En la sección 3 se describe y se valida formalmente el enfoque de verificación composicional que sustenta nuestro EFVC. En la sección 4 se presenta la integración de los elementos que permiten la verificación de un MT utilizando el EFVC. En la sección 5 se describe la aplicación a un ejemplo de MPN relacionado con el *Manejo de las Relaciones con el Cliente* (conocido como CRM). Por último, en la sección 6, se presentan las conclusiones y se describe brevemente el trabajo futuro.

## 2. Comportamientos de un MT bajo un mismo dominio semántico

Se considera que CSP+T es el lenguaje formal apropiado para describir de forma no ambigua a los participantes en los PN si se quiere capturar su comportamiento temporal y la secuencia de *comunicación* en la que están involucrados. CSP+T [4] es un lenguaje de especificación de tiempo real que, como extensión del cálculo de procesos *Communicating Sequential Processes*

(CSP) [8], nos permite interpretar el efecto de ejecutar un constructo sintáctico considerando *sólo un estado a la vez* en la secuencia de ejecución y de esta forma poder calcular las transiciones [8] a las cuales están sujetos los participantes. Los operadores de CSP+T relacionados con el tiempo son [4]: (a) el evento especial de instanciación de procesos denotado  $\star$  (estrella); (b) el operador de captura de tiempo ( $\triangleright$ ) asociado con la función  $\alpha_e = s(e)$ , que guarda en la variable  $\alpha$  (variable marcadora) el instante de aparición del evento  $e$  (evento marcado) cuando éste se manifiesta; y (c) el intervalo de habilitación de un evento, denotado como  $I(T, t_1).e$ , el cual establece que el evento  $e$  debe ocurrir dentro del intervalo de tiempo  $[t_1, t_1+T]$  para que sea interpretado por el término de proceso en el que está programado. Todos estos operadores representan descripciones temporizadas más detalladas de sistemas especificados inicialmente como no temporizados y facilita la especificación y prueba de propiedades temporales de sistemas [4]. El uso de CSP+T añade los modelos de trazas y fallos a la verificación composicional de PN, los cuales son fundamentales para decidir si el modelo que expresa el comportamiento de los PN satisface la especificación de las propiedades no funcionales. Además, gracias al concepto de *refinamiento con el tiempo* [8], logramos verificar el comportamiento temporal especificado en CSP+T considerando como condición necesaria la secuencia de eventos del modelo no temporizado de CSP.

### 2.1. Modelo de un MT

Partiendo de la semántica de las entidades de BPMN que se analizan en [2], se aplican las reglas de transformación que presentamos en la Tabla 1 y se hace corresponder cada entidad BPMN (objetos de flujo, objetos de conexión, y swimlanes) [1] con un término sintáctico de proceso secuencial CSP+T que especifica el comportamiento de los participantes. Esta formalización extiende la capacidad descriptiva de las entidades de modelado BPMN introduciendo restricciones temporales con el objeto de dar mayor expresividad y precisión a los constructos de BPMN que dependen del tiempo. El conjunto completo de reglas se presenta en [9].

En resumen, las transformaciones detalladas en la Tabla 1 se realizan mediante la asigna-

ción de: (1) cada una de las entidades de modelado BPMN a un término prefijo de CSP+T, (2) cada tiempo de duración discreto a un intervalo de habilitación de un evento, y (3) se utiliza la elección externa para especificar las alternativas que el entorno de los procesos, cuya composición representa a los PN, puede ejercer sobre los términos de proceso CSP+T para garantizar su terminación cuando acabe la ejecución del MT.

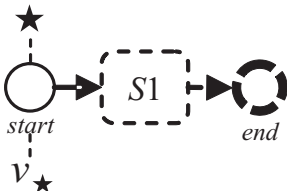
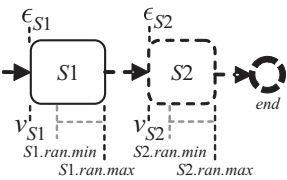
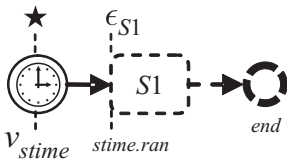
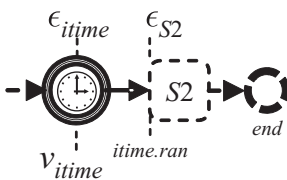
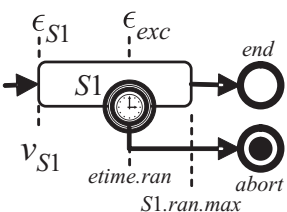
### 2.2. Propiedades de un MT

Para especificar las propiedades que un MT debe exhibir, se utiliza la lógica temporal de intervalos denominada *Concurrent Computation Tree Logic* (CCTL) [10] que nos permite llevar a cabo un razonamiento lógico a nivel de intervalos de tiempo, y no en instantes de tiempo. Sin embargo, para poder llevar a cabo la verificación de un MT, las propiedades deben expresarse en el mismo dominio semántico del modelo. Para ello, se aplica: (1) el algoritmo que se describe en [11], el cual permite construir un *Autómata de Büchi Temporizado* (ABT) discreto semánticamente equivalente a una fórmula CCTL, y (2) el procedimiento descrito en [12], el cual permite transformar el ABT obtenido anteriormente en términos de procesos de CSP+T. La construcción que obtenemos de (1) se fundamenta en un procedimiento iterativo que, tomando en cuenta el estado en el cual se encuentra el autómata, junto con el siguiente operador y su tiempo de aceptación, obtiene determinísticamente el próximo cambio de estado. Mientras que la transformación que se realiza con (2) se basa en reglas de transformación que permiten obtener los términos de procesos CSP+T partiendo de la estructura de datos del ABT generada en (1).

## 3. Verificación de un MT: el EFVC

La instanciación del EFVC [3] se basa en el hecho de que un MT  $C$  se ha estructurado en varios PN  $C_i$  (participantes) verificados, que trabajan en paralelo,  $C = \parallel_{i:1..n} C_i$ , donde cada participante  $C_i$  satisface la propiedad  $\phi_i$ , lo que representa la especificación del comportamiento que cada proceso debe exhibir. El principal objetivo es hacer posible la verificación del comportamiento del MT a partir de la verificación del com-

Tabla 1  
Algunas reglas de correspondencia entre BPMN y CSP+T

Elemento BPMN	Descripción	Proceso CSP+T
	<p>El inicio (<i>start</i>) corresponde al evento especial de instanciación <math>\star</math> de CSP+T y la variable marcadora <math>v_\star</math> es usada para almacenar el instante de ocurrencia del evento <math>\star</math>.</p>	$P(\text{start}) = (\star \gg v_\star \rightarrow \text{SKIP}; P(\text{start}))$ $\square(\epsilon_{\text{end}} \rightarrow \text{SKIP})$
	<p>La actividad S1 antecede a la actividad S2. La actividad S1 comienza cuando ocurre el evento <math>\epsilon_{S1}</math> y la invocación de la actividad S2 (lo cual corresponde a la ocurrencia del evento <math>\epsilon_{S2}</math>) debe ocurrir dentro del intervalo de tiempo <math>[S1.ran.min, S1.ran.max]</math>.</p>	$P(S1) = (\epsilon_{S1} \gg v_{S1} \rightarrow \text{SKIP}; I(S1.ran.max - S1.ran.min, v_{S1} + S1.ran.min). \epsilon_{S2} \rightarrow \text{SKIP}; P(S1))$ $\square(\epsilon_{\text{end}} \rightarrow \text{SKIP})$
	<p>El inicio temporizado (<i>stime</i>) establece que la actividad S1 debe comenzar (lo cual corresponde a la ocurrencia del evento <math>\epsilon_{S1}</math>) <i>stime.ran</i> unidades de tiempo después de la ocurrencia del evento especial de instanciación <math>\star</math>.</p>	$P(\text{stime}) = (\star \gg v_{\text{stime}} \rightarrow \text{SKIP}; I(\text{stime.ran}, v_{\text{stime}}) \rightarrow \text{SKIP}; \epsilon_{S1} \rightarrow \text{SKIP}; P(\text{stime}))$ $\square(\epsilon_{\text{end}} \rightarrow \text{SKIP})$
	<p>De acuerdo con el intermedio temporizado (<i>itime</i>), la actividad S2 debe comenzar (lo cual corresponde a la ocurrencia del evento <math>\epsilon_{S2}</math>) <i>itime.ran</i> unidades de tiempo después de la ocurrencia del evento <math>\epsilon_{itime}</math>.</p>	$P(\text{itime}) = (\epsilon_{\text{itime}} \gg v_{\text{itime}} \rightarrow \text{SKIP}; I(\text{itime.ran}, v_{\text{itime}}) \rightarrow \text{SKIP}; \epsilon_{S2} \rightarrow \text{SKIP}; P(\text{itime}))$ $\square(\epsilon_{\text{end}} \rightarrow \text{SKIP})$
	<p>La ejecución de la actividad S1 puede ser interrumpida (lo cual corresponde a la ocurrencia del evento <math>\epsilon_{exc}</math>) <i>etime.ran</i> unidades de tiempo después de su inicio (lo cual corresponde a la ocurrencia del evento <math>\epsilon_{S1}</math>) y antes del final de la duración de la actividad (es decir, dentro del intervalo de tiempo <math>[v_{S1}, S1.ran.max]</math>).</p>	$P(S1) = (\epsilon_{S1} \gg v_{S1} \rightarrow \text{SKIP}; I(S1.ran.max - S1.ran.min, v_{S1} + S1.ran.min). \epsilon_{\text{end}} \rightarrow (\text{SKIP} \Delta I(\text{itime.ran}, v_{S1}) \rightarrow \text{SKIP}); \epsilon_{\text{exc}} \rightarrow \text{SKIP}; \text{abort} \rightarrow \text{STOP}; P(S1))$ $\square(\epsilon_{\text{end}} \rightarrow \text{SKIP})$

Legenda:  $\epsilon_x$  denota la invocación de las entidades de modelado BPMN [1],  $Sx.ran.min$  y  $Sx.ran.max$  denotan el tiempo mínimo y máximo, respectivamente, de duración de las actividades  $Sx$ ; y, *stime.ran* e *itime.ran* denotan la demora definida por el inicio temporizado (*stime*) y el intermedio temporizado (*itime*), respectivamente.

portamiento de sus PN locales. Por ello, cada proceso  $C_i$  debe satisfacer el *invariante* ( $\psi_i$ ), el cual representa el comportamiento de los otros participantes del modelo de tareas con respecto a  $C_i$ . Se usa el símbolo especial  $\neg\delta$  para denotar que no puede producirse el bloqueo del participante (es decir, que el proceso caiga en un estado sin ninguna transición de salida) en ninguna ejecución posible en la que colabore. La propiedad  $\phi$  y el invariante  $\psi$  que son satisfechos por el MT  $C$  son obtenidos a partir de las propiedades locales  $\phi_i$  ( $\wedge_{i:1..n} \phi_i \Rightarrow \phi$ ) y de los invariantes locales  $\psi_i$  ( $\wedge_{i:1..n} \psi_i \Rightarrow \psi$ ), respectivamente. Como resultado, podemos obtener la verificación completa del MT utilizando el Teorema 1<sup>1</sup>:

**Teorema 1**  
**(Verificación composicional de un MT)**

Sea el MT  $C$  estructurado en varios participantes que trabajan en paralelo,  $C = \parallel_{i:1..n} C_i$ . Para un conjunto de términos de procesos CSP+T( $C_i$ ) en el que cada uno describe el comportamiento de los participantes  $C_i$ , y satisfacen las propiedades  $\phi_i$ , los invariantes  $\psi_i$ , y la ausencia de bloqueo  $\neg\delta$ , con  $\cap_{i:1..n} \Sigma_i = \emptyset$ ,  $\cap_{i:1..n} \Omega_i = \emptyset$ , y  $\cap_{i:1..n} \Gamma(T(C_i)) = \emptyset$ , entonces la siguiente condición se cumplirá:

$$T(C) \models (\phi \wedge \psi \wedge \neg\delta) \Leftrightarrow \parallel_{i:1..n} T(C_i) \models \wedge_{i:1..n} (\phi_i \wedge \psi_i) \wedge \neg\delta, \tag{1}$$

donde  $T(C) = \parallel_{i:1..n} T(C_i)$ .

La aplicación de la relación (1) incluye realizar una verificación del *chequeo de satisfacción* inductivo en el rango de términos de procesos CSP+T de los componentes ( $i:1..n$ ) del MT. Esta verificación es automatizada en nuestro enfoque mediante la integración de la herramienta FDR2 [13].

En la Figura 1 se muestra nuestra visión del proceso de verificación de un MT.

Partiendo del Diagrama de Proceso de Negocio de BPMN, teniendo en consideración las reglas y metas del negocio, la instanciación del EFVC para un MT consta de los procesos integrados (ver Figura 1) descritos a continuación.

**Modelado del MT**

La descripción completa del comportamiento del MT es modelado por el término de procesos de CSP+T denotado como  $T(C)$ , el cual es *interpretado* a través de un conjunto de términos de proceso CSP+T, denotados  $T(C_i)$ , usando la propuesta de semántica temporal para las enti-

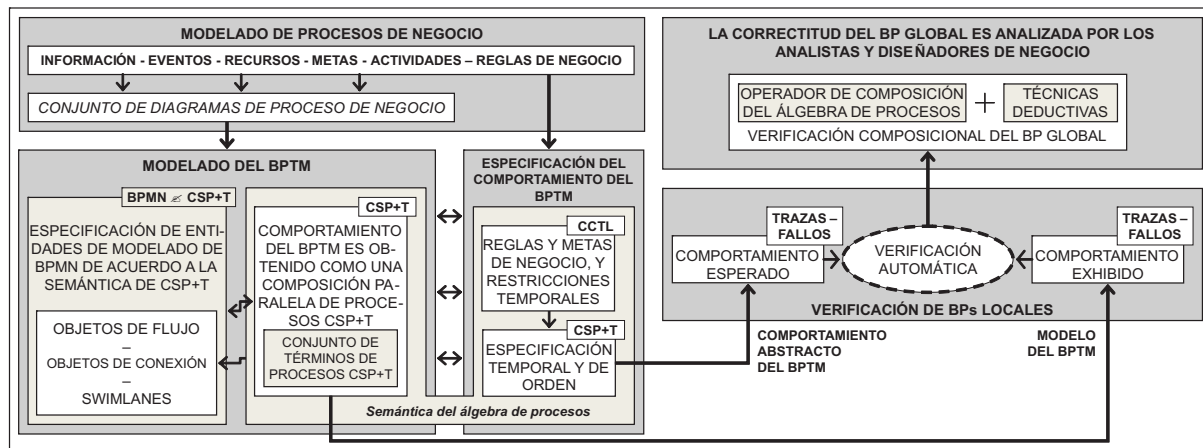


Figura 1. Visión integrada de la verificación composicional de un MT.

1 Denotamos  $T(C) \models \phi$  cuando el  $T(C)$  mantiene la propiedad  $\phi$  para todos los estados iniciales que representa, es decir,  $\forall s \in I \Rightarrow (T(C), s) \models \phi$ . Para las invariantes tenemos  $T(C) \models \psi$  cuando para todos los  $s \in S$  se mantiene  $(T(C), s) \models \psi$ . Para denotar que  $T(C)$  no contiene algún bloqueo, escribimos  $T(C) \models \neg\delta$ .

dades de modelado de BPMN introducida en la sección 2.1.

### Especificación del comportamiento del MT

Por medio de fórmulas CCTL son *especificados* los requisitos y las restricciones temporales que el MT debe cumplir. Posteriormente, estas propiedades son expresadas por términos de procesos CSP+T llamados  $T(\phi_i)$ ,  $T(\psi_i)$  y  $T(-\delta)$ , siguiendo los algoritmos citados en la sección 2.2.

### Verificación

Se procede a verificar el comportamiento del MT, ejecutando los siguientes pasos:

1. Los procesos locales  $T(C_i)$  son verificados de forma automática respecto del conjunto de procesos  $T(\phi_i)$ ,  $T(\psi_i)$  y  $T(-\delta)$ , de acuerdo con las semánticas de trazas y de fallos de los cálculos de procesos basados en CSP. Para ello, se procede a introducir en la herramienta FDR2 [13] las siguientes expresiones:

$$T(\phi_i) \sqsubseteq_T T(C_i) \wedge T(\psi_i) \sqsubseteq_T T(C_i) \wedge T(-\delta) \sqsubseteq_T T(C_i) \quad (2)$$

$$T(\phi_i) \sqsubseteq_F T(C_i) \wedge T(\psi_i) \sqsubseteq_F T(C_i) \wedge T(-\delta) \sqsubseteq_F T(C_i) \quad (3)$$

Con lo anterior, FDR2 verifica automáticamente la correctitud del comportamiento de los participantes (o PN locales) de acuerdo con las siguientes relaciones:

- Relativas a aspectos de seguridad:

$$\begin{aligned} \forall t \in \text{traces}(T(\phi_i)) \exists t' \in \text{traces}(T(C_i)) : \\ t' \Rightarrow \phi_i \Leftrightarrow T(C_i) \models \phi_i \end{aligned} \quad (4)$$

$$\begin{aligned} \forall t \in \text{traces}(T(\psi_i)) \exists t' \in \text{traces}(T(C_i)) : \\ t' \Rightarrow \psi_i \Leftrightarrow T(C_i) \models \psi_i \end{aligned} \quad (5)$$

$$\begin{aligned} \forall t \in \text{traces}(T(-\delta)) \exists t' \in \text{traces}(T(C_i)) : \\ t' \Rightarrow -\delta \Leftrightarrow T(C_i) \models -\delta \end{aligned} \quad (6)$$

- Relativas a aspectos de vivacidad:

$$\begin{aligned} \forall (t, X) \in \mathcal{SF}[T(\phi_i)] \exists (t', X) \in \mathcal{SF}[T(C_i)] : \\ (t', X) \Rightarrow \phi_i \Leftrightarrow T(C_i) \models \phi_i \end{aligned} \quad (7)$$

$$\begin{aligned} \forall (t, X) \in \mathcal{SF}[T(\psi_i)] \exists (t', X) \in \mathcal{SF}[T(C_i)] : \\ (t', X) \Rightarrow \psi_i \Leftrightarrow T(C_i) \models \psi_i \end{aligned} \quad (8)$$

$$\begin{aligned} \forall (t, X) \in \mathcal{SF}[T(-\delta)] \exists (t', X) \in \mathcal{SF}[T(C_i)] : \\ (t', X) \Rightarrow \neg \delta \Leftrightarrow T(C_i) \models \neg \delta \end{aligned} \quad (9)$$

Por último, mediante la aplicación del Teorema 1, se obtiene la verificación completa del comportamiento del MT  $T(C)$  de acuerdo con la relación (1).

## 4. Ejemplo de aplicación

Para mostrar la aplicabilidad de la propuesta, ésta fue utilizada en un proyecto empresarial de MPN relacionado con el negocio denominado CRM. CRM es una estrategia que permite lograr y mantener la relación entre una empresa y sus clientes, ofreciéndoles una atención individual y, a su vez, participación en la relación comercial [14]. Sólo se muestra un ejemplo de aplicación de la semántica temporal propuesta para BPMN, por lo que se centrará en los PN locales involucrados en el proceso *Vender Producto/Servicio* (Proc\_Vender\_P/S), debido a su importancia para la estrategia de CRM. La información necesaria para llevar a cabo un razonamiento formal acerca de los PN se muestra en el diagrama del proceso Vender Producto/Servicio que se presenta en la Figura 2, el cual estipula la sincronización de los participantes para llevar a cabo las tareas involucradas en los flujos de mensajes.

### 4.1. Definición y descripción del MT

Para obtener la especificación del proceso Vender Producto/Servicio en términos de procesos CSP+T, de acuerdo a la propuesta descrita en la sección 3.1, definimos los conjuntos  $CU$  y  $CO$  para indexar los procesos que corresponden a los elementos notacionales de los participantes *Cliente* (denotado como *Cus*) y *Compañía* (denotado como *Com*), respectivamente (ver Figura 2).

$$CU = \{\text{start.1, cu\_s1, cu\_s2, cu\_s3, cu\_s4, cu\_s5, cu\_s6, xgate.1, end.1, abort.1}\} \quad (10)$$

$$CO = \{\text{start.2, co\_s1, co\_s2, co\_s21, co\_s3, co\_s4, co\_s5, co\_s6, co\_s7, co\_s8, agate.1, agate.2, end.2, abort.2}\} \quad (11)$$

$$Cus = \text{let } X = \square i : (\alpha Y \setminus \{\text{fin.1, abt.1}\}) \bullet \\ (i \rightarrow X \square \text{fin.1} \rightarrow \text{SKIP} \square \text{abt.1} \rightarrow \text{STOP})$$

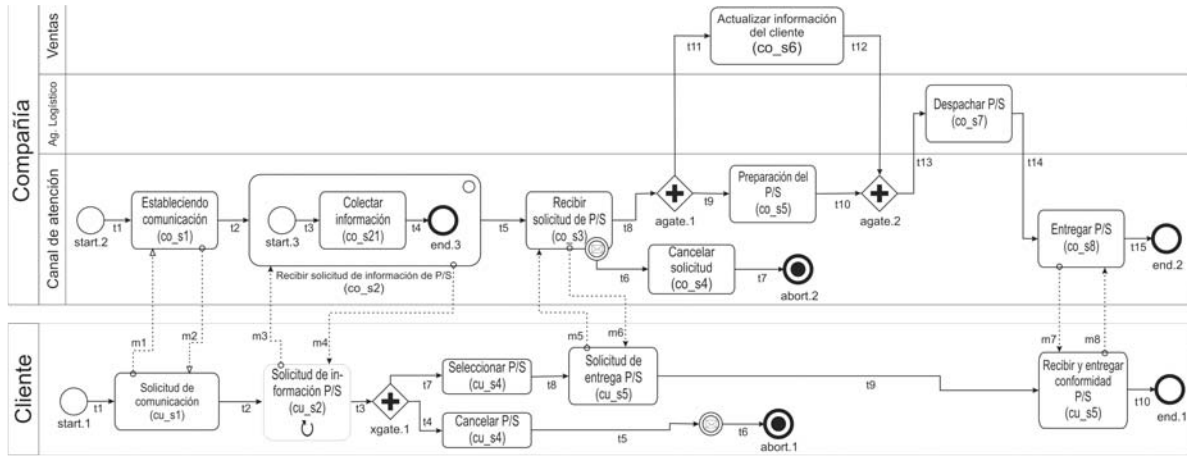


Figura 2. Diagrama del proceso de negocio *Vender Producto/Servicio*.

$$Y = \langle \langle i: CU \bullet \alpha P(i) \circ P(i) \rangle \text{ within} \\ (Y \parallel \alpha Y \parallel X) \setminus \{ \text{init.Cus} \} \rangle \quad (12)$$

$$\text{Com} = \text{let } Z = \square j: (\alpha R \setminus \{ \text{fin.2}, \text{abt.2} \}) \bullet \\ (j \rightarrow Z \square \text{fin.2} \rightarrow \text{SKIP} \square \text{abt.2} \rightarrow \text{STOP}) \\ R = \langle \langle j: CO \bullet \alpha P(j) \circ P(j) \rangle \text{ within} \\ (R \parallel \alpha R \parallel Z) \setminus \{ \text{init.Com} \} \rangle \quad (13)$$

donde por cada  $i \in CU$  y  $j \in CO$ , los procesos  $P(i)$  y  $P(j)$ , respectivamente, son definidos a continuación. Debido a limitaciones de espacio sólo mostraremos algunos de los términos de proceso que componen a  $Cus$  y a  $Com$  para ilustrar la aplicación de la semántica propuesta<sup>2</sup>.

$$P(\text{start.1}) = (t0, \star \rightarrow \text{init.Cus.cu_s1} \rightarrow \text{SKIP}) \square \\ \text{fin.1} \rightarrow \text{SKIP} \quad (14)$$

$$P(\text{co_s3}) = (\text{init.Com.co_s3} \succ \text{vs3} \rightarrow \text{SKIP}; \\ \text{starts.Com.co_s3} \rightarrow (\text{SKIP} \triangle \\ (l(600, \text{vs3}).\text{msg.co_s3?x:}\{\text{cancel}\} \rightarrow \\ \text{SKIP}; \text{init.Com.co_s4} \rightarrow \text{SKIP}) \square \\ (\text{msg.co_s3!x:}\{\text{in,last}\} \rightarrow \text{SKIP}; \\ \text{msg.co_s3.out} \rightarrow \text{SKIP}; \\ l(600, \text{vs3}).\text{init.Com.agate.1} \rightarrow \text{SKIP}; \\ P(\text{co_s3})) \square \text{fin.2} \rightarrow \text{SKIP} \quad (15)$$

$$P(\text{end.2}) = \text{init.Com.end.2} \rightarrow \text{SKIP}; \text{fin.2} \rightarrow \text{SKIP} \quad (16)$$

La colaboración entre los participantes *Cliente* y *Compañía* es la composición paralela de los procesos  $Cus$  y  $Com$ , tal como es denotado por el término de procesos CSP+T  $Proc\_Vender\_P/S$ :

$$Proc\_Vender\_P/S = (Cus \parallel [\alpha Cus \parallel \alpha Com]) \parallel Com \\ \setminus \{ \text{msg} \} \quad (17)$$

El término de procesos  $Proc\_Vender\_P/S$  conforma el MT del proceso de negocio *Vender Producto/Servicio* que verificamos con respecto a las propiedades que se presentan a continuación.

#### 4.2. Definición de propiedades

Para mostrar la aplicación de la propuesta, se trabajará con la propiedad: *una vez que el Cliente inició una comunicación con la Compañía existe la obligación de hacerle llegar el Producto/Servicio y se ha de recibir la confirmación de la entrega*. Como se hará la verificación del comportamiento del MT  $Proc\_Vender\_P/S$  a partir del comportamiento de los participantes  $Cus$  y  $Com$ , entonces éstos deben ejecutar todas sus actividades en el orden esperado para lograr que el proceso funcione globalmente. Las propiedades parciales que se deben verificar en los PN locales  $Cus$  y  $Com$ , respectivamente, para obtener la verificación del proceso  $Proc\_Vender\_P/S$ , son:

2 Expresamos los tiempos de duración en segundos de acuerdo a la función  $sec$  definida en [2].





2. Los conjuntos de etiquetado de ambos componentes,  $\Gamma(Cus)$  y  $\Gamma(Com)$ , son disjuntos, que también pueden ser comprobado como sigue:

$$\Gamma(Cus) \cap \Gamma(Com) = \emptyset \quad (26)$$

$$\begin{aligned} \Gamma(Cus) &= \{\text{start.1, cu\_s1, cu\_s2, cu\_s3, cu\_s4,} \\ &\quad \text{cu\_s5, cu\_s6, xgate.1, end.1, abort.1}\} \wedge \\ \Gamma(Com) &= \{\text{start.2, co\_s1, co\_s2, co\_s21, co\_s3,} \\ &\quad \text{co\_s4, co\_s5, co\_s6, co\_s7,} \\ &\quad \text{co\_s8, agate.1, agate.2, end.2, abort.2}\} \end{aligned} \quad (27)$$

Por consiguiente, después de haber verificado que las relaciones (22), (24), y (26), son verdaderas, podemos concluir que *Cus* y *Com* pueden *componerse*. Por el Teorema 1, tenemos:

$$\begin{aligned} (Cus \parallel [\alpha Cus \parallel \alpha Com] \parallel Com) \setminus \{\text{msg}\} \models \\ \phi_{Cus} \wedge \phi_{Com} \end{aligned} \quad (28)$$

y como  $Proc\_Vender\_P/S = (Cus \parallel [\alpha Cus \parallel \alpha Com] \parallel Com) \setminus \{\text{msg}\}$  y  $\phi_{Proc\_Vender\_P/S} = \phi_{Cus} \wedge \phi_{Com}$ , obtenemos:

$$Proc\_Vender\_P/S \models \phi_{Proc\_Vender\_P/S} \quad (29)$$

En resumen, se ha obtenido la verificación del proceso *Proc\_Vender\_P/S*, que representa el comportamiento del MT del proceso de negocio Vender Producto/Servicio, a partir de los procesos individuales verificados *Cus* y *Com*.

## 5. Conclusiones

En este trabajo se ha presentado y validado la adaptación del EFVC para la especificación y verificación del comportamiento de un modelo de tareas derivado de los PN a ser soportados mediante un SI, a partir de la verificación de forma independiente del comportamiento de los participantes individuales que realizan los PN. La propuesta modela los participantes del proceso como términos de procesos sintácticos de CSP+T, ya que este lenguaje formal de especificación de sistemas soporta la composición sintáctica de los términos de proceso a través del operador de composición concurrente. Como complemento del EFVC, también se propone una semántica

temporal para BPMN en términos de CSP+T, lo que permite especificar los tiempos de respuesta de las actividades y las tareas en los modelos desarrollados; así como las restricciones temporales referidas a la comunicación y la colaboración entre tareas, y el intervalo de tiempo válido para captura de los flujos de excepción, de acuerdo con el comportamiento esperado de los PN. Adicionalmente, se ha mostrado el valor potencial y la viabilidad de nuestro enfoque a través de su aplicación a un ejemplo de interés empresarial que incluye requisitos temporales de colaboración relacionado con el área de CRM. Utilizando la herramienta FDR2 para verificar la correctitud de los términos de procesos de CSP+T que describen el comportamiento de los participantes principales de los procesos, se logró la verificación composicional del modelo de tareas. El trabajo actual y a mediano plazo se centrará en la aplicación del EFVC y la semántica temporal propuesta para BPMN a casos de estudio relacionados con organizaciones reales, donde existan una vasta cantidad de procesos operativos que contengan tareas temporizadas, con la finalidad de estudiar la factibilidad del enfoque para negocios complejos.

## Referencias bibliográficas

1. OMG: "Business Process Model and Notation-v2.0", Object Management Group, Massachusetts, USA, 2011.
2. Wong, P. y Gibbons, J.: "A Process Semantics for BPMN". Lecture Notes in Computer Science, Vol. 5256 (2008) 355-374.
3. Capel, M., Mendoza, L. y Benghazi, K.: "Automatic Verification of Business Process Integrity". International Journal on Simulation and Process Modelling, Vol. 4, No. 3/4 (2008), 167-182.
4. Žic, J.: "Time-constrained Buffer Specifications in CSP+T and Timed CSP". ACM Transaction on Programming Languages and Systems, Vol. 16 (1994) 1661-1674.
5. Ma, S., Zhang, L. y He, J.: "Towards Formalization and Verification of Unified Business Process Model based on Pi Calculus. Proc. ACIS International Conference on Software Engineering Research, Management and Applications 1, (2008) 93-101.

6. Aalst, W.: "Making Work Flow: On the Application of Petri Nets to Business Process Management". Lecture Notes in Computer Science, Vol. 2360 (2002) 1-22.
7. Morimoto, S.: "A Survey of Formal Verification for Business Process Modeling". Lecture Notes in Computer Science, Vol. 5102 (2005) 514-522.
8. Schneider, S.: "Concurrent and Real-Time Systems-The CSP Approach", John Wiley & Sons, Ltd., Chichester, England, 2000.
9. Mendoza, L., Capel, M. y Pérez, M.: "A Formalization Proposal of Timed BPMN for Compositional Verification of Business Processes". Lecture Notes in Business Information Processing, Vol. 73, Part 4 (2011) 388-403.
10. Rűf, J. y Kropf, T.: "Symbolic Model Checking for a Discrete Clocked Temporal Logic with Intervals". Proc. of the IFIP WG 10.5 International Conference on Correct Hardware Design and Verification Methods, (1997) 146-163.
11. Mendoza, L. y Capel, M.: "Algorithm Proposal to Automata Generation from CCTL Formulas". Technical report UGR-SC-2008-01, University of Granada, 2008. Disponible en: [http://prof.usb.ve/lmendoza/Documentos/TR\\_UGR-SC-2008-01.pdf](http://prof.usb.ve/lmendoza/Documentos/TR_UGR-SC-2008-01.pdf).
12. Mendoza, L. y Capel, M.: "Procedure Proposal to CSP+T process terms Generation from Automata". Technical report UGR-SC-2009-01, University of Granada, 2009. Disponible en: [http://prof.usb.ve/lmendoza/Documentos/TR\\_UGR-SC-2009-01.pdf](http://prof.usb.ve/lmendoza/Documentos/TR_UGR-SC-2009-01.pdf).
13. Formal Systems (Europe) Ltd.: "Failures-Divergence Refinement-FDR2 User Manual", Formal Systems (Europe) Ltd., Oxford, UK, 2005.
14. Mendoza, L., Marius, A., Pérez, M. y Grimán, A.: "Critical Success Factors for a Customer Relationship Management Strategy". Information Software Technology, Vol. 49, No. 8 (2007) 913-945.

Recibido el 26 de Enero de 2011

En forma revisada el 19 de Noviembre de 2012