

## **SCK: Una ontología para evaluar la performance de una cadena de suministro en ambientes de simulación distribuida**

Juan L. Sarli<sup>1</sup>, Ma. de los Milagros Gutiérrez<sup>2</sup>

<sup>1</sup> INGAR, Instituto de Diseño y Desarrollo CONICET – UTN, Santa Fe, Argentina  
juanleonardosarli@santafe-conicet.gov.ar

<sup>2</sup> CIDISI, Centro de Investigación y Desarrollo de Ingeniería en Sistemas de Información UTN,  
Santa Fe, Argentina  
mmgutier@frsf.utn.edu.ar

**Abstract.** Utilizar simulación distribuida para analizar cadenas de suministro favorece el reuso de simuladores independientes pertenecientes a los miembros de la cadena minimizando el costo de desarrollo de un simulador. Sin embargo, componer estos simuladores es un problema a resolver. Existen diferentes niveles de composición de simuladores: sintáctico, semántico y pragmático. Para atacar este problema es necesario expresar el dominio y el conocimiento de los componentes de una manera no ambigua y estandarizada. En este contexto, las ontologías son útiles ya que proveen una forma de representar el conocimiento a través de taxonomías de conceptos, axiomas, reglas y relaciones entre estos. Se presenta la ontología SCK como parte de una red de ontologías, para representar el conocimiento semántico de una cadena de suministro basada en los conceptos principales del modelo SCOR con el fin de proveer una herramienta adecuada para evaluar cadenas de suministro en el contexto de simulación distribuida.

**Keywords:** Simulación distribuida, HLA, cadena de suministro, modelo SCOR, ontologías.

### **1 Introducción**

En los últimos quince años la comunidad de modelado y simulación ha demostrado un creciente interés hacia la construcción de modelos de simulación a partir de la composición de modelos existentes [1-4]. Este esquema es sumamente conveniente ya que promueve el reuso de los simuladores utilizados por cada uno de los participantes, minimizando el tiempo y la inversión de construcción de la simulación, y además, preserva tanto la autonomía local como la privacidad de los datos [5]. Estas características hacen adecuada la utilización de este enfoque para simular cadenas de suministro (CS). Una CS es una de las redes organizacionales más populares donde sus miembros realizan alianzas para alcanzar metas más importantes de las que obtendrían de forma aislada [6], [7]. El éxito de una CS depende de la coordinación

de las actividades de los participantes para hacer eficiente el flujo de materiales, de información y financiero. En este contexto, la simulación de una CS adquiere un rol fundamental.

La construcción de modelos de simulación a partir de otros modelos, denominados “bloques” reutilizables, trae aparejado el problema de la composición de simuladores. Dicho problema posee diferentes niveles, los cuales son: sintáctico, semántico y pragmático. La composición sintáctica se refiere a la conexión y comunicación entre componentes. La composición semántica determina si el modelo de simulación, armado a partir de los bloques, es semánticamente válido. Este tipo de composición determina si los bloques que integran el sistema de simulación están compuestos de una manera significativa y dicha composición es válida [8], [9]. Para componer sistemas de simulación correctamente y alcanzar una interoperabilidad válida la alineación y comprensión consistente debe ser alcanzada a nivel de un modelo conceptual [10]. Finalmente, la composición pragmática establece si los componentes son conscientes del contexto de simulación en el que se están ejecutando [3], y de ser este el caso, los componentes conocen cual es el uso que se le va a dar a los datos [11].

HLA (High Level Architecture) [12] es una propuesta ampliamente usada para construir simuladores a partir de la composición de simuladores independientes que interactúan a través de una interface común, para ello propone un esquema de federaciones y federados. Esta propuesta es una solución a la composición sintáctica de simuladores ya que estandariza el protocolo de comunicación y el formato del modelo de objetos que debe ser utilizado por los componentes. Para dar soporte a la composición semántica de simuladores en este esquema es necesario expresar el dominio de los componentes de una forma no ambigua, donde la semántica de los conceptos sea representada en forma explícita.

En este contexto, las ontologías son utilizadas para organizar la representación del conocimiento y capturar objetos de información en un dominio particular [13]. Continuando con esta investigación, en [5] se presenta la red de ontologías *SCFHLA* (Supply Chain Federation HLA) como propuesta para dar soporte a la composición semántica de simuladores en el contexto de una federación de CS. Este modelo conceptual considera cuatro dominios específicos: (i) El dominio de las CS, el cual es modelado por la ontología *SCOnto* (Supply Chain Ontology) [14] basada en el modelo *SCOR* (Supply Chain Operation Reference) [15], (ii) el dominio de federaciones modelado por la ontología *FEDOnto* (Federation Ontology), (iii) el dominio del modelo de objetos *BOMOnto* (Base Object Model Ontology) y (iv) el dominio de los modelos de empresa *EMOnto* (Enterprise Model Ontology).

Sin embargo, esta propuesta es una definición de alto nivel que muestra una aproximación a la resolución del problema. A su vez, *SCOnto* está basada en una versión previa del modelo *SCOR* y algunos de sus conceptos deben ser actualizados.

Por lo tanto para superar las debilidades señaladas, este trabajo propone incluir la ontología *SCK* (Supply Chain Knowledge) en la red *SCFHLA* de manera que dicha red (con la ontología *SCK* añadida) sea la base conceptual de una herramienta que permita la generación semiautomática de un modelo de objetos conceptual para una instancia dada de una federación HLA. El objetivo principal de la ontología propuesta

es representar el conocimiento semántico de una CS usando la versión actual del modelo SCOR y relacionarlo con los conceptos básicos de una federación HLA de simulación para evaluar la performance de una CS.

El resto del trabajo se organiza de la siguiente manera. En la siguiente sección se presentan los conceptos preliminares para entender el trabajo tales como el modelo SCOR y el estándar HLA. Luego se presenta la ontología SCK y su integración a la red SCFHLA. Seguido se desarrolla un ejemplo donde se muestra la utilización de la ontología propuesta. Finalmente el trabajo se concluye y los trabajos futuros son presentados.

## 2 Conceptos Preliminares

### 2.1 Modelo SCOR

Este modelo fue desarrollado por la organización Supply Chain Council, y presenta los conceptos más importantes para modelar una CS. Además, permite comparar y evaluar el rendimiento de la CS como un todo y de sus actividades particulares. Dado que SCOR es un modelo conceptual, provee una terminología común y facilita el entendimiento a través de la CS. Este modelo ayuda a analizar, medir, fijar objetivos de performance, determinar oportunidades de mejora, identificar mejores prácticas y priorizar proyectos en la gestión de la CS.

SCOR [15] está organizado alrededor de los seis procesos de gestión básicos: Plan, Source, Make, Deliver, Return y Enable. También, define tres niveles jerárquicos de procesos los cuales son:

- Nivel 1 o Alcance: Define los tipos de procesos, alcance y contenido de la CS.
- Nivel 2 o Configuración: Define las categorías de procesos y la estrategia de operaciones.
- Nivel 3 o Pasos: Define los elementos de proceso y la configuración de los procesos individuales.

En todos los niveles, el modelo provee indicadores claves de rendimiento, los cuales se dividen sistemáticamente en cinco atributos de performance: Confiabilidad, Capacidad de Respuesta, Agilidad, Costos y Gestión de Activos (Activos).

De acuerdo al modelo SCOR un atributo de rendimiento es un conjunto de métricas usadas para expresar una estrategia [15]. En el contexto del modelo SCOR un atributo no posee el mismo significado que en modelado conceptual. Un atributo por sí mismo no puede ser medido; es usado para determinar una dirección estratégica. Las métricas de SCOR están organizadas en una estructura jerárquica. El modelo describe métricas de nivel uno, dos y tres. Las relaciones entre estos niveles son diagnósticas, por ejemplo: las métricas de nivel 2 sirven como diagnóstico para las de nivel 1. Esto significa que observando el rendimiento de las métricas de nivel 2 se puede explicar las diferencias de performance, o las posibles mejoras, para las métricas de nivel 1. Este tipo de análisis de rendimiento de la CS se conoce como descomposición de métricas [15].

## 2.2 Estándar HLA

HLA es una arquitectura estándar para reuso, integración e interoperabilidad de simuladores. Soporta el reuso de capacidades disponibles en diferentes simuladores, y posibilita el desarrollo de simuladores complejos de manera distribuida. La arquitectura HLA provee un framework dentro del cual los desarrolladores de simuladores pueden estructurar y describir sus aplicaciones de simulación. En este sentido HLA propone integrar sistemas de simulación diferentes, en un sistema mayor sin necesidad de reescribir algún componente o comenzar a crear desde cero el modelo de mayor nivel y su respectivo simulador. Cuando un simulador se implementa para que cumpla las especificaciones HLA se lo llama *Federado*. Luego, los federados se unen en federaciones las que se refieren a la simulación del sistema complejo formado. La definición de HLA abarca tres componentes esenciales (Figura 1) [12]:

- **Reglas:** Es un conjunto de ítems que definen las responsabilidades y relaciones entre los componentes de una federación HLA. Deben ser seguidas por los federados y las federaciones que quieran ajustarse al estándar. Seguir estas reglas asegura una interacción correcta entre los simuladores en una federación.
- **Especificación de la Interfaz:** Define la interfaz funcional entre federados HLA y la infraestructura de ejecución (RTI - Run-Time Infrastructure) de HLA. Se detallan los servicios que debe prestar el RTI, e identifica las funciones callback que debe proveer cada federado. El RTI ofrece una librería de programación y una interfaz de programación de aplicaciones (API – Application Program Interface) compatible con la especificación de la interfaz.
- **Patrón de Modelo de Objeto (OMT):** Provee un framework común para la comunicación entre los federados. OMT consta de los siguientes documentos:
  - Modelo de Objetos de la Federación (FOM - Federation Object Model): Describe los objetos, los atributos y las interacciones de la federación completa.
  - Modelo de Objetos de la Simulación (SOM - Simulation Object Model): Describe los objetos, atributos e interacciones usados por un federado.
  - Modelo de Objetos de Gestión (MOM - Management Object Model): Provee facilidades para acceder a información operativa del RTI en tiempo de ejecución. El MOM debe ser definido utilizando objetos, interacciones y constructores de la arquitectura de comunicación de HLA.

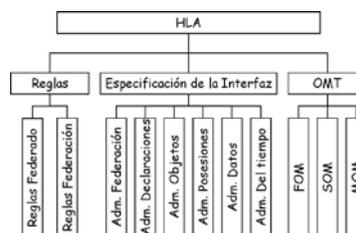


Fig. 1. Componentes HLA

Cada federado tiene un SOM que describe los datos que éste puede producir o consumir. Cada federación tiene un FOM que describe las partes comunes de los SOM correspondientes a los federados que participan y que serán usados en la federación. El RTI es el software necesario para ejecutar la federación y básicamente consiste en una implementación de la especificación de la interfaz compuesta por un conjunto de servicios. El RTI provee funcionalidades para comenzar y terminar la ejecución de la simulación, transferir datos entre los simuladores intervinientes, controlar la cantidad de datos que son transferidos y finalmente, coordinar el paso del tiempo de simulación entre los simuladores HLA. Este software está fuera del alcance de la especificación de HLA y es suministrado por proveedores de software específicos.

### 3 Ontología SCK

La ontología SCK se desarrolla con el objetivo de dar soporte a la evaluación de la performance en una CS a través de los conceptos del modelo SCOR. Esta ontología es integrada a la red de ontologías SCFHLA aportando de esta manera un mecanismo de evaluación y medición en el contexto de simulación distribuida de CS.

En la Figura 2 se presenta la red de ontologías SCFHLA, en la que se muestra la ontología SCK y su integración a la red a través de la meta-relación *evaluates* que vincula SCK con la ontología FEDOnto siendo la meta-relación inversa *isEvaluated-By*.

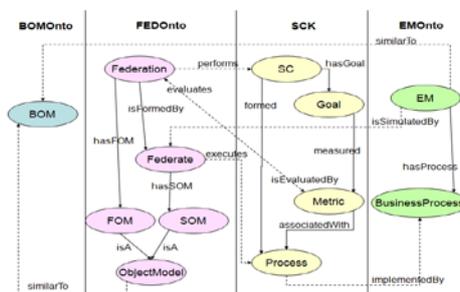


Fig. 2. Red SCFHLA

Según las prácticas recomendadas por la IEEE DSEEP (Distributed Simulation Engineering and Execution Process) [16] la primer tarea para desarrollar una federación es definir los objetivos de la misma. Dado que la federación se encuentra en el dominio de las CS, al definir un objetivo éste se traduce en un atributo de performance según lo establecido por el modelo SCOR. No es posible medir por sí mismo un atributo de performance, por lo que es necesario elegir un conjunto de métricas para evaluar el grado en que se logra cumplir con el mismo. Las métricas seleccionadas necesitan cierta información para calcular sus mediciones la cual es provista por los federados. Para establecer qué información necesita y genera un federado es imprescindible definir el FOM. Esta tarea normalmente implica una búsqueda de recursos reutilizables o la generación manual del mismo. Por lo tanto,

con el objetivo de lograr una generación semiautomática del FOM se presenta la red SCFHLA, con especial interés en la ontología SCK, la que modela los conceptos de una CS por medio del modelo SCOR. Para el desarrollo de la ontología se utiliza principalmente la metodología NeON [17], empleando además de la metodología METHONTOLOGY la actividad de evaluación de ontologías [18].

En la Figura 3 se presenta, por medio de un diagrama de clases UML, la ontología SCK y su relación con la ontología FEDOnto.

Se define el concepto *SupplyChain* como el concepto más general que representa la CS que se quiere simular. Dado que SCOR propone definir procesos que conforman la cadena, estos procesos se modelaron a través del concepto *Process* el cual tiene una subclase *SubProcess* para identificar que un proceso puede descomponerse en otros procesos más simples. A su vez estos procesos estarán relacionados unos con otros, a través del concepto *Relation* que representa los flujos de información y material que existe entre los procesos. *SupplyChain* tiene una meta denominada *Goal* la cual se encuentra asociada al concepto *PerformanceAttribute* que conceptualiza los atributos de performance definidos en SCOR. Estos atributos de performance señalan la dirección estratégica de la CS y se miden por medio de una métrica, que está representada por el concepto *Metric*. A su vez, el concepto *Metric* está asociada al concepto *Process* identificando que un *Process* tiene asociada una o más métricas y que una métrica puede estar asociada a varios procesos diferentes. Tanto *Process* como *Metric* tienen un atributo *level* que identifica el nivel de acuerdo con los niveles definidos en SCOR. Para poder calcular una métrica se necesita una fórmula. Así se define en la ontología el concepto *Formula*, que está asociado a *Metric*. El atributo *result* de *Formula* identifica el resultado que arroja su evaluación, el atributo *calculation* identifica la manera en que la fórmula será calculada y el atributo *unit* identifica la unidad a utilizar en el cálculo (por ej. día, mes, orden, etc). Una *Formula* posee *Variable* la que puede ser *AtomicVariable* o *ComplexVariable*. Una *AtomicVariable* es una medición de un recurso mientras que una *ComplexVariable* está compuesta de una o varias *Metric* ya que representa una medida de alto nivel sobre un recurso.

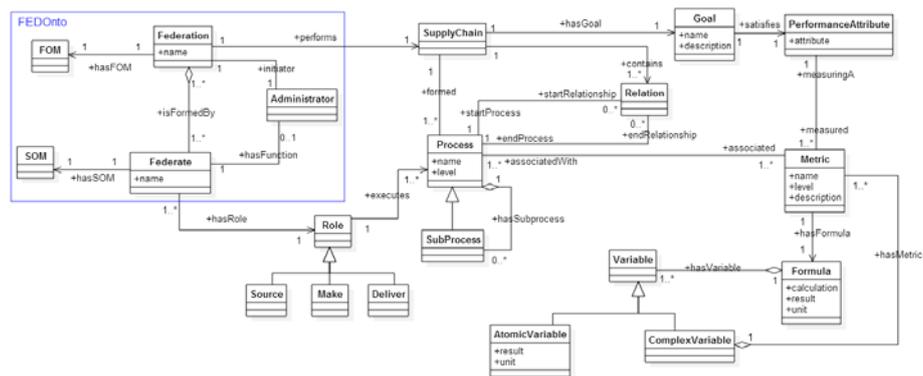


Fig. 3. Ontología SCK y FEDOnto

En la ontología *FEDOnto* se representan los conceptos relacionados con la simulación distribuida de acuerdo al modelo HLA. Así se representa la federación con el concepto *Federation* y el federado se representa por el concepto *Federate*. Para este contexto particular una federación representa la simulación de una CS. Para su representación se definió la relación *performs* que relaciona el concepto *Federation* con *SupplyChain*. Además se identifica que un federado participante de una federación debe tener un rol asociado para identificar el proceso de la CS que dicho federado modela. Para su representación, se definieron los conceptos *Role* que identifica el rol y tres subclases del mismo *Source*, *Make* o *Deliver*. A su vez, se define la relación *executes* entre *Role* y *Process* para indicar que un federado que juega ese rol ejecuta ese proceso de la cadena.

La ontología SCK se implementó utilizando la herramienta Protégé<sup>1</sup> versión 4.3. Para transformar el modelo conceptual de la Figura 3 en la ontología se aplicaron los siguientes criterios: (a) Las clases se transformaron en clases OWL<sup>2</sup> (Web Ontology Language). (b) Las relaciones se mapearon como Object Properties. (c) Los atributos se tradujeron en Data Properties.

#### 4 Definición de una Federación de CS

En esta sección se presenta un ejemplo de uso de la ontología SCK. Considere una CS cuya meta es: *Cumplir de forma perfecta con el 80% de las entregas de autos desde la fábrica hacia el concesionario durante el período de un año*. Esta CS se crea como una instancia de *SupplyChain* denominada Automóvil CS en la ontología. Su meta se define como instancia de *Goal* cuyo nombre es Objetivo CS y que contiene en el atributo *description* el objetivo de la CS. Según el modelo SCOR, esta meta se traduce en el atributo de performance “Confiabilidad” y es medido por la métrica de nivel uno “Cumplimiento Perfecto de Orden”. Este atributo de performance se denomina Atributo CS y es una instancia de *PerformanceAttribute* cuya propiedad *attribute* contiene el atributo “Confiabilidad”. La métrica que lo releva se nombra como Ordenes Perfectas y es una instancia de *Metric* que posee el atributo *level* en uno y la propiedad *name* con el nombre de dicha métrica. El cumplimiento perfecto de orden posee una fórmula que se crea como instancia de *Formula* denominada Formula Ordenes Perfectas. Además, esta fórmula posee en el atributo *calculation* como calcularse y en el atributo *unit* su unidad de medida. Para el cálculo de las órdenes perfectas se necesitan dos variables: total de órdenes y total de órdenes perfectas. La primera se define en el modelo como instancia de *AtomicVariable* con el mismo nombre y con la unidad orden. La segunda se denomina como Total Ordenes Perfectas y es instancia de *ComplexVariable*. Esta variable compleja se compone de cuatro métricas de nivel dos las cuales son: 1) Ordenes Entregadas Completas, 2) Entrega en Fecha, 3) Documentación Precisa y 4) Condición Perfecta. Cada una de ellas se crea como instancia de *Metric*. La métrica 1) posee la fórmula denominada Formula Orden Completa que es instancia de *Formula*. Dicha fórmula tiene las *AtomicVariable* Ordenes Entregadas

<sup>1</sup> <http://protege.stanford.edu/>

<sup>2</sup> [http://semanticweb.org/wiki/OWL\\_2](http://semanticweb.org/wiki/OWL_2)

y Ordenes Entregadas Completas. La métrica 2) tiene la formula nombrada Formula Entrega en Fecha que es instancia de *Formula*. Esta última contiene las *AtomicVariable* Ordenes Entregadas y Ordenes Entregadas en Fecha. La métrica 3) cuenta con la formula llamada Formula Documentación Precisa que es instancia de *Formula*. Esta fórmula posee las *AtomicVariable* Ordenes Entregadas y Ordenes Entregadas Documentación Exacta. La métrica 4) es calculada por medio de la fórmula que se denomina Formula Condición Perfecta que es instancia de *Formula*. Dicha fórmula tiene las *AtomicVariable* Ordenes Entregadas y Ordenes Entregadas Condición Perfecta.

Para generar la simulación distribuida de la CS Automóvil CS, el primer paso es crear una federación y definir su objetivo. Entonces, se crea AutomóvilCSH como instancia de *Federation* con los participantes fábrica de automóviles y concesionario de autos. Se establece como meta: “Analizar durante el periodo de un año las entregas de automóviles de la fábrica al concesionario.” Durante este tiempo de simulación, se analizaran las entregas para determinar si se alcanza con éxito el objetivo de la CS definido anteriormente.

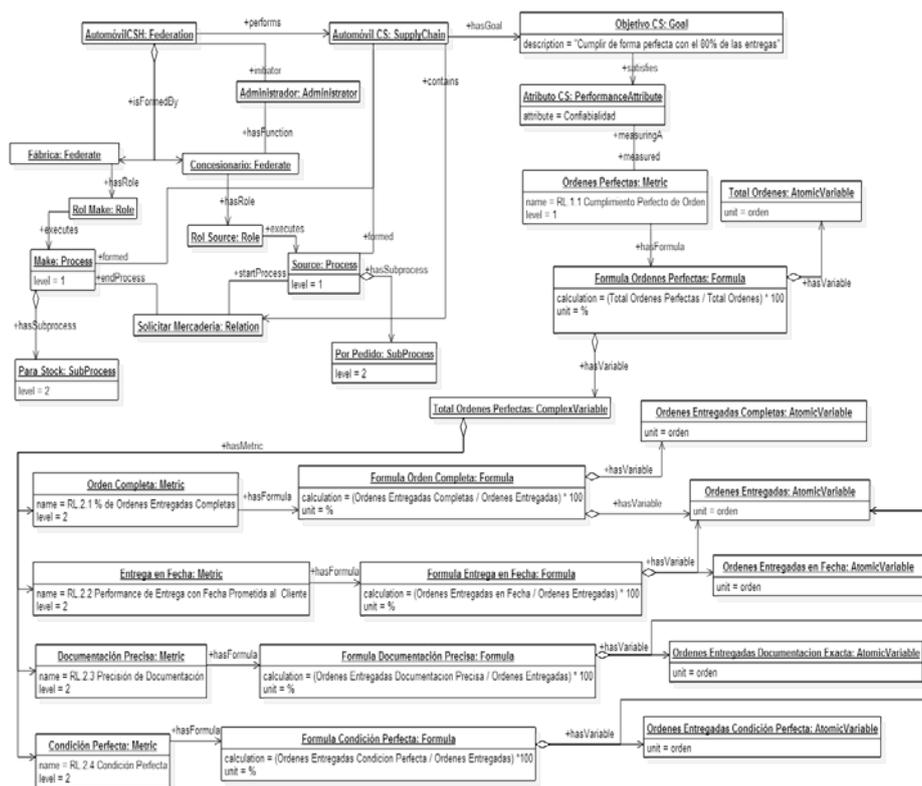


Fig. 4. Instanciación de la ontología SCK

Luego, los miembros se añaden a la federación y deben estar de acuerdo en la definición del FOM. Dado que los participantes son dos, se crean los federados Fábrica y

Concesionario como instancias de *Federate*. Se asigna el Rol Make a la fábrica porque transforma la materia prima en los autos, en tanto que, se establece el Rol Source al concesionario porque son de interés las interacciones que tiene con la fábrica para obtener los autos. Ambos roles se crean como instancias de *Role*. La Fábrica por medio del Rol Make tiene asociado un proceso de producción para poder construir los autos. Este proceso se denomina Make (instancia de *Process*) y posee un subproceso Para Stock (instancia de *Subprocess*). El Concesionario por medio del Rol Source cuenta con un proceso para solicitar el envío de autos. Este proceso se nombra Source (instancia de *Process*) y cuenta con un subproceso Por Pedido (instancia de *Subprocess*). La relación entre el proceso Source y Make se denomina Solicitar Mercadería y se crea como instancia de *Relation*. En la Figura 4 se muestran los conceptos instanciados para este ejemplo.

## 5 Conclusión

Este trabajo ha presentado la ontología SCK para dar soporte a la evaluación de la performance de una CS mediante los conceptos del modelo SCOR. Esta ontología se ha integrado a la red de ontologías SCFHILA, permitiendo de este modo, evaluar y medir una simulación distribuida de CS. Como trabajo futuro se pretende avanzar en nuevos conceptos, relaciones, propiedades, axiomas y reglas para representar con un mayor nivel de detalle el dominio. Algunas de las propiedades que se pueden añadir al modelo son: identificador de procesos, métricas y de tipos de procesos. Este último puede ser utilizado para definir reglas lógicas que validen la composición proceso subproceso. Por lo tanto, es posible validar que un proceso este compuesto solo de subprocesos del mismo tipo.

Dada la modularidad de la red de ontologías, se está trabajando en implementar una nueva ontología para el dominio de federaciones e integrarla a la red, para lograr una representación más detallada del dominio. A su vez, la red SCFHILA con la ontología SCK integrada sirve de base para el desarrollo de una herramienta para definir una federación de forma colaborativa y que siga los pasos descritos en DSEEP.

## Referencias

- [1] P. L. Gustavson and L. M. Root, "Object model use cases: a mechanism for capturing requirements and supporting BOM reuse," *Spring Simulation Interoperability Workshop*, Mar. 1999.
- [2] S. Kasputis and H. C. Ng, "Composable simulations," in *Simulation Conference, 2000. Proceedings. Winter, 2000*, vol. 2, pp. 1577–1584 vol.2.
- [3] A. Tolk, "What Comes After the Semantic Web - PADS Implications for the Dynamic Web," in *20th Workshop on Principles of Advanced and Distributed Simulation, 2006. PADS 2006*, Beach Road, Singapore, 2006, pp. 55–55.
- [4] A. Verbraeck, "Component-based distributed simulations: the way forward?," in *18th Workshop on Parallel and Distributed Simulation, 2004. PADS 2004*, Kufstein, Austria, 2004, pp. 141–148.

- [5] M. Milagros Gutiérrez and Horacio Leone, “Composability Model In A Distributed Simulation Environment For Supply Chain,” presented at the 42° JAIIO Simposio Argentino de Informática Industrial (SII), Córdoba, Argentina, 2013, pp. 142–153.
- [6] W. W. C. Chung, A. Y. K. Yam, and M. F. S. Chan, “Networked enterprise: A new business model for global sourcing,” *International Journal of Production Economics*, vol. 87, no. 3, pp. 267–280, Feb. 2004.
- [7] J. Arrazola, “Towards a new model: the globally integrated enterprise,” *Universia Business Review*, vol. 14, pp. 108–118, May 2007.
- [8] E. W. Weisel, M. D. Petty, and R. Mielke, “Validity of models and classes of models in semantic composability,” in *Proceedings of the Fall Simulation Interoperability Workshop*, 2003.
- [9] M. D. Petty, E. W. Weisel, and R. Mielke, “Overview of a Theory of Composability,” presented at the Virginia Modeling Analysis & Simulation Center, 2004.
- [10] M. A. Hofmann, “Challenges of Model Interoperation in Military Simulations,” *SIMULATION*, vol. 80, no. 12, pp. 659–667, Dec. 2004.
- [11] B. P. Zeigler and P. E. Hammonds, *Modeling & Simulation-Based Data Engineering: Introducing Pragmatics into Ontologies for Net-Centric Information Exchange*, 1 edition. Burlington, MA: Elsevier Academic Press, 2007.
- [12] Institute of Electrical and Electronics Engineers and IEEE-SA Standards Board, *IEEE standard for modeling and simulation (M & S) high level architecture (HLA) framework and rules*. New York: Institute of Electrical and Electronics Engineers, 2010.
- [13] Asunción Gómez-Pérez, Mariano Fernández-López, and Oscar Corcho, *Ontological Engineering*. London: Springer-Verlag, 2004.
- [14] A. C. Böhm, H. P. Leone, and G. P. Henning, “A Supply Chain Ontology Conceptualization with Focus on Performance Evaluation,” in *Proceedings of the Tenth International Conference on Enterprise Information Systems*, Portugal, 2008, vol. ISAS-2, pp. 402–409.
- [15] Supply Chain Council, *SCOR supply chain operations reference model*. The Supply Chain Council, 2012.
- [16] Institute of Electrical and Electronics Engineers and IEEE-SA Standards Board, *IEEE recommended practice for distributed simulation engineering and execution process (DSEEP)*. New York: Institute of Electrical and Electronics Engineers, 2011.
- [17] M. C. Suárez-Figueroa, A. Gómez-Pérez, and M. Fernández-López, “The NeOn Methodology for Ontology Engineering,” in *Ontology Engineering in a Networked World*, M. C. Suárez-Figueroa, A. Gómez-Pérez, E. Motta, and A. Gangemi, Eds. Springer Berlin Heidelberg, 2012, pp. 9–34.
- [18] Karin Koogan Breitman, Marco Antonio Casanova, and Walter Truszkowski, “Semantic Web: Concepts, Technologies, and Applications,” *Computer*, vol. 40, no. 7, pp. 84–84, Jul. 2007.