

Análisis de Flujos de Conocimiento en Proyectos de Mejora de Procesos Software bajo una perspectiva multi-enfoque

Brenda L. Flores-Rios ¹, Francisco J. Pino ², Jorge E. Ibarra-Esquer ³, Félix Fernando González-Navarro ¹, Oscar M. Rodríguez-Elías ⁴

brenda.flores@uabc.edu.mx, fjpino@unicauca.edu.co, jorge.ibarra@uabc.edu.mx, fernando.gonzalez@uabc.edu.mx, omrodriguez@ith.mx

¹ Instituto de Ingeniería. Universidad Autónoma de Baja California, Blvd. Benito Juárez s/n Col. Insurgentes Este, C.P. 21280, Mexicali, Baja California, México.

² Grupo IDIS, Facultad de Ingeniería Electrónica y Telecomunicaciones, Universidad del Cauca Calle 5 No. 4 – 70. Popayán, Cauca, Colombia.

³ Facultad de Ingeniería, campus Mexicali. Blvd. Benito Juárez s/n. Col. Insurgentes Este. C.P. 21280. Mexicali, Baja California, México.

⁴ División de Estudios de Posgrado e Investigación. Instituto Tecnológico de Hermosillo. Ave. Tecnológico y Periférico Poniente s/n. Col. Sahuaro. C.P. 83170. Hermosillo, Sonora. México.

DOI: 10.17013/risti.14.51-66

Resumen: La Gestión de conocimiento (GC) tiene el potencial de proveer diversos beneficios a las organizaciones dedicadas al desarrollo de software. Entre ellos la gestión de activos de conocimiento, el aumento de la productividad o el incremento de la capacidad de sus procesos software. Sin embargo, debido al crecimiento y evolución que ha tenido la industria de software, se requieren nuevos paradigmas asociados a la generación y reutilización de diversos tipos y flujos de conocimiento con el propósito de promover procesos de innovación centrados en la Mejora de procesos software (SPI). En este artículo se presenta un análisis de tres enfoques de flujos de conocimiento, con el cual se ofrece una estrategia para apoyar la creación, transferencia y evolución del conocimiento involucrado en un proyecto SPI. Además, se define un marco de trabajo para la identificación de flujos de conocimiento relacionados con elementos de un modelo de referencia de procesos.

Palabras-clave: Flujos de Conocimiento; Procesos Software; Proyecto de Mejora de Procesos Software.

Analysis of Knowledge Flows in Software Process Improvement Projects under a multi-perspective approach

Abstract: Knowledge Management (KM) can provide several benefits to software development organizations. Some of them are knowledge assets management, productivity increase, or an improvement in their software process capability level.

However, due to the evolution and growth in the software development industry, there is a requirement for new paradigms associated to creation and reuse of different types and flows of knowledge, tending to encourage innovation processes around Software Process Improvement (SPI). This paper presents an analysis of three approaches to knowledge flows, proposing a strategy that supports the creation, exchange and evolution of knowledge involved in an SPI project. We define a framework for capture of different types of knowledge and identification of knowledge flows from the elements of a process reference model.

Keywords: Knowledge Flows; Software Process; Software Process Improvement Project;

1. Introducción

En las dos últimas décadas, se han definido técnicas de software, modelos de procesos, normas y/o estándares internacionales apropiados para enfrentar las exigencias de la industria de software en un entorno globalizado y competitivo (Pino, García & Piattini, 2007; Oktaba & Piattini, 2008). La necesidad de las organizaciones dedicadas al desarrollo de software, en gran parte VSEs (por sus siglas en inglés de Very Small Entities), de generar ventajas competitivas sostenibles en el tiempo, las ha llevado a definir estrategias en la gestión de sus activos de conocimiento debido a que éstos se relacionan con procesos de creación, estructuración y transferencia de conocimiento (Capote *et al.*, 2008). Una estrategia que puede ser útil es implementar un proyecto de Mejora de procesos software (SPI por sus siglas en inglés de Software Process Improvement) basado en conocimiento que tome en cuenta los procesos, productos, recursos y roles orientados al conocimiento; así como la identificación y caracterización de los activos de conocimiento (experiencias, lecciones aprendidas, mejores y buenas prácticas) en función de promover la mejora continua al enfatizar la generación y utilización de diversos tipos y flujos de conocimiento (Flores-Ríos *et al.*, 2014).

Por lo anterior, se considera de relevancia que en un proyecto de SPI se gestionen los activos de conocimiento y se identifiquen los tipos y flujos de conocimiento existentes en una organización, para alcanzar el nivel de competitividad deseado o incrementar la capacidad de sus procesos. De esta forma, cada ciclo de mejora se considera como un proceso continuo y evolutivo (Mathiassen & Pourkomeylian, 2003) que genera, combina y reutiliza tanto tipos, activos y procesos de conocimiento con el propósito de mejorar la manera de realizar las actividades o resolver determinados problemas. Se ha observado que algunos modelos no brindan herramientas que permitan identificar o extraer el conocimiento que poseen los individuos de las VSEs, lo cual es fundamental para la formalización de las actividades (Carvajal & Márquez, 2013). En este sentido, el objetivo de este artículo es presentar un marco de trabajo para la identificación de tipos y flujos de conocimiento asociado a un modelo de referencia de procesos que una VSE dedicada al desarrollo de software pueda utilizar en un proyecto SPI. La característica principal del marco de trabajo es que integra un marco conceptual de Gestión de Conocimiento (GC) en procesos software bajo una perspectiva de flujos de conocimiento y presenta, en un modelo declarativo, los procesos de ciclo de vida de GC asociados a una dimensión del proceso. La implicación práctica del marco es que puede ser utilizado en investigaciones de GC en procesos software, así como en trabajos

relativos al diseño o implementación de proyectos SPI basados en conocimiento dirigidos a VSEs.

Este documento se estructura de la siguiente manera: en la sección 2 se presenta trabajo relacionado con la aplicación de la Gestión de conocimiento en el área de Mejora de procesos software y se describen tres enfoques centrados en flujos de conocimiento como modelos que apoyan la identificación, estructura y relación entre los componentes de la organización. La sección 3 detalla las fases de la metodología KoFI presentando los resultados generales asociados al flujo de conocimiento existente en las entidades proceso software utilizados en un proyecto SPI. En la sección 4 se presentan las aportaciones e implicaciones prácticas de los elementos obtenidos en esta investigación para la realización de un par de iniciativas SPI en el contexto de la industria del software. Por último, se exponen las conclusiones.

2. Contextualización

En un proyecto de Mejora de procesos software (SPI) es necesario involucrar un modelo que conduzca la mejora, un modelo de referencia de procesos a seguir y un método para la evaluación de procesos (Pino *et al.*, 2006). El modelo que gestiona la mejora describe la infraestructura, actividades, ciclo de vida y consideraciones prácticas para guiar la iniciativa de SPI en la organización. El modelo de referencia de procesos describe cuáles actividades son reconocidas como las mejores y buenas prácticas que una organización debe implementar para la producción de software (Oktaba & Piattini, 2008). Por último, el modelo de evaluación de procesos especifica la ejecución de un método de evaluación formal, principalmente cuantitativo, para conocer la capacidad del proceso o la madurez de la organización. La interrelación de los modelos necesarios para llevar a cabo un proyecto SPI se presenta en la Figura 1.

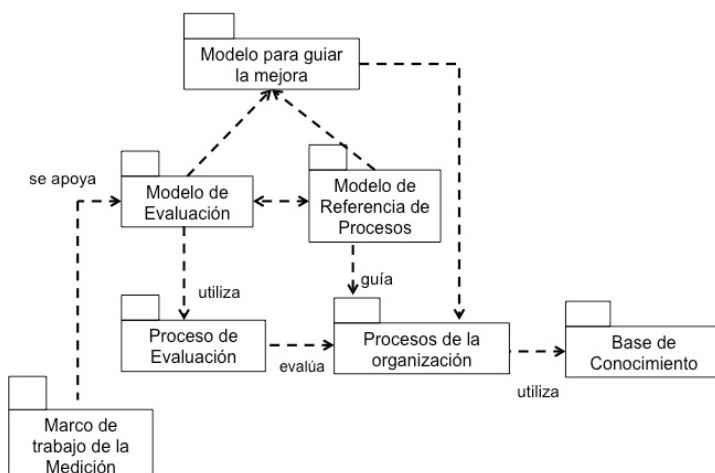


Figura 1 – Interrelación de elementos que conforman un proyecto de Mejora de procesos software

Es importante indicar que este trabajo se centra en analizar los tipos y flujos de conocimiento existentes en la descripción de procesos, en función de un modelo de

referencia de procesos. Según el estándar de propósito general ISO/IEC 24774:2007 – *Ingiería de software y sistemas – Administración de ciclo de vida – Guía para la descripción de procesos*, los elementos primarios involucrados en la descripción de un proceso, en cualquier modelo de referencia de procesos, son: título, propósito, actividades y salidas (ISO/IEC TR 24774, 2007). Estos se representan de manera explícita y formal por medio de plantillas, patrones de procesos o marcos descriptivos y son necesarios en la evaluación de procesos (Kruchten, 1998; NMX-I-059, 2011). Los objetivos de la formalización son hacer visibles los procesos de la organización, disminuir la dependencia hacia un rol, replicar buenas prácticas en la realización de actividades y/o procesos e institucionalizar procesos de calidad y mejora continua (Carvajal & Márquez, 2013). Desde un enfoque de GC, el principal objetivo que debe tener una estrategia es facilitar y mejorar el flujo de conocimiento para que se aplique cuando, dónde y por quién lo requiera. Así, el primer paso que debe considerarse al proponer estrategias para proyectos SPI basados en conocimiento dentro de una VSE, es analizar la forma en la que el conocimiento fluye o se gestiona dentro de la organización.

2.1 Trabajo relacionado

Diversos trabajos se han realizado con el objetivo de analizar la forma en la que las organizaciones dedicadas al desarrollo de software gestionan su conocimiento, o para proponer mecanismos, estrategias o sistemas de GC (Rodríguez-Elías & Martínez García, 2011). Este tipo de organización se considera como una entidad intensiva en conocimiento por lo que la transferencia de conocimiento es crítica para ella. Kautz y Nielsen (2004) desarrollaron un marco de trabajo práctico que ayuda a los agentes de cambio a entender la implementación de un proyecto SPI como un proceso de transferencia del conocimiento. Además, existen trabajos que brindan evidencia de la aplicación de la GC en los procesos de desarrollo de software enfocándose en la reutilización de activos de conocimiento para mejorar la calidad de los productos actuales o nuevos, las fases de ciclo de vida del software o facilitar la reutilización del conocimiento explícito (Niazi, Wilson & Zowghi, 2005; Anaya, Cechich & Henao, 2007; Gopesh, Ward & Mohan, 2007; Wan *et al.*, 2011). Se encontraron pocos trabajos relacionados con facilitar y mejorar los flujos de conocimiento como apoyo en la transferencia de capacidad y experiencia de donde reside a donde se necesita a través del tiempo, el espacio y distribución geográfica (Nissen & Levitt, 2002). Esto supone un inconveniente debido a que los flujos de conocimiento representan el atributo dinámico del conocimiento (Guo & Wang, 2008) y la transferencia del *saber cómo*, la cual consiste en un valor estratégico para una organización (Gupta & Govindarajan, 2000). Por tal motivo, el aporte de este trabajo es analizar bajo una perspectiva multi-enfoque los flujos de conocimiento que intervienen en gran medida en la implementación de un proyecto SPI.

2.2 Enfoques de Flujos de conocimiento

De acuerdo con Rodriguez-Elías & Martínez García (2011) existen tres enfoques centrados en flujos de conocimiento, los cuales se describen en las siguientes secciones.

Enfoque 1. Procesos de creación y transferencia de conocimiento.

Está basado en diferenciar los tipos de conocimiento Táctico y Explícito y los cuatro procesos asociados a su conversión (Nonaka & Takeuchi, 1995). El conocimiento tácito se refiere al conocimiento construido por individuos, lo que implica que es difícil de comunicar y transferir, imitar o medir porque está fundamentado en las relaciones humanas o en hábitos comunes (Garzón Castrillón & Fisher, 2008). A diferencia del conocimiento tácito, el conocimiento explícito es tangible. En las organizaciones de software, el conocimiento explícito se identifica en forma de procedimientos y procesos escritos, productos de trabajo, archivos, bases de datos, artículos de investigación y difusión, libros de normas o estándares, código, mensajes de correo electrónico, entre otros (Flores-Ríos, Rodríguez-Elías & Pino, 2013).

A partir de la interacción, dinámica y continua, entre el conocimiento tácito y explícito se crea la espiral de conocimiento representada en el modelo SECI (Figura 2a). El objetivo del modelo es brindar un entendimiento de cómo las organizaciones crean conocimiento para maximizar su administración, aplicación y transferencia por medio de cuatro procesos de creación y transferencia de conocimiento (Nonaka & Takeuchi, 1995): Socialización, Exteriorización, Combinación, e Interiorización.

Enfoque 2. El ciclo de conocimiento.

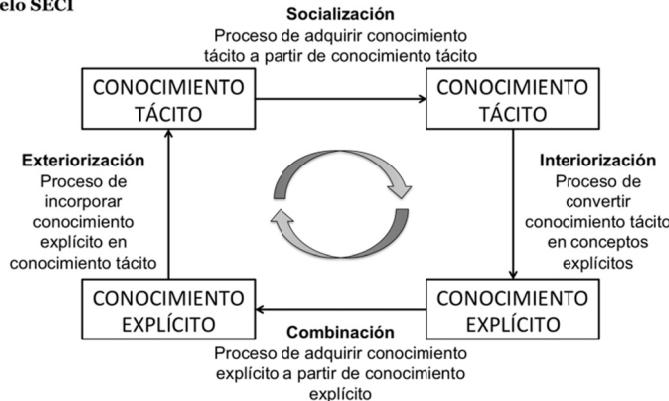
En este enfoque, el conocimiento es visto como un recurso que brinda la capacidad de usar información y aplicarla en la solución de un problema, realización de alguna tarea o toma de decisión (Carlsson, 2002). En el ciclo de conocimiento (Choo, 1999) se identifican la percepción del medio ambiente, la creación del conocimiento y la acción como toma de decisiones (Figura 2b). La percepción se obtiene por el uso de diversas fuentes de conocimiento que sirven para los flujos de experiencias. La percepción conduce a la creación de significados compartidos o modelos mentales que son utilizados para planear y tomar decisiones. Las conexiones entre los tres elementos representan el conocimiento requerido y generado por las actividades, así como las fuentes de información o conocimiento de donde éste es obtenido o almacenado (Rodríguez-Elías & Martínez García, 2011).

Enfoque 3. Modelo integrado para el ciclo de vida de Gestión de conocimiento.

El modelo integrado propuesto por Rodríguez-Elías & Martínez García (2011) se apoya en los procesos de transferencia y conversión del conocimiento tácito y explícito, donde ambos tipos de conocimiento son mutuamente complementarios apoyando la interacción dinámica entre los procesos de conocimiento que realizan los roles. Así mismo, son dinámicos en función a la etapa de ciclo de vida de GC en la que se encuentren. En el inciso c de la Figura 2, se visualiza la interacción de la creación y adquisición del conocimiento con la aplicación del mismo. La exteriorización es llevada por las actividades de captura, organización y codificación para que pueda ser almacenado. La transferencia de conocimiento explícito es habilitada por un conjunto de actividades para su formalización, almacenamiento y posterior recuperación. Finalmente, la interiorización requiere de la recuperación del conocimiento para filtrar e interpretar lo que realmente es de utilidad para quien lo aplica.

A partir de la investigación realizada, se hizo evidente la transdisciplinariedad entre la GC y la SPI y se generó un marco teórico de los tres enfoques centrados en flujos de conocimiento. Dichos enfoques son parte de una estrategia de implementación de un proyecto SPI basado en conocimiento y se integran en las fases de la metodología KoFI (por sus siglas en inglés de Knowledge Flow Identification).

a) Modelo SECI



b) Ciclo de conocimiento



c) Modelo integrado para el ciclo de vida de GC

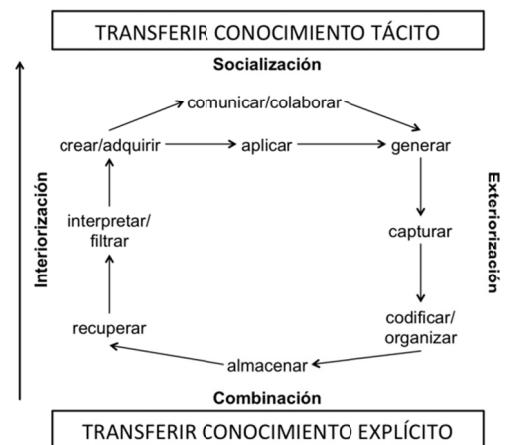


Figura 2 – Enfoques centrados en flujos de conocimiento. a) Modelo SECI, b) Ciclo de conocimiento y c) Modelo Integrado para el ciclo de vida de Gestión de Conocimiento

3. Metodología para el análisis de flujos de conocimiento

La metodología KoFI define los lineamientos para la identificación y el análisis de flujos de conocimiento en los procesos organizacionales y su aplicación es un proceso iterativo e incremental que retroalimenta o regresa de una fase a cualquier otra. Así mismo, se compone de las siguientes fases (Rodríguez-Elías & Martínez García, 2011): 1) especificar el proceso a analizar, 2) modelar el proceso con enfoque en flujos de conocimiento, 3) identificar flujos y tipos de conocimiento, y 4) identificar la forma en que el conocimiento fluye dentro del grupo de trabajo. A continuación, se presentan algunos de los resultados obtenidos al ejecutar cada una de las cuatro fases.

3.1. Especificar el proceso a analizar.

En la industria de software, es posible establecer un conjunto de entidades proceso software que deben estar presentes en todos los procesos del proyecto SPI. De esta forma, las entidades proceso software que se utilizan en este trabajo están relacionadas al conjunto de actividades que se realizan para el logro del propósito del proceso, los recursos requeridos (roles, recursos de software y hardware), los productos de trabajo de entrada y salida, los procedimientos adoptados (métodos, técnicas y plantillas) y el modelo de ciclo de vida que será usado (Falbo & Bertollo, 2009).

3.2. Modelar el proceso con enfoque en flujos de conocimiento.

KoFI especifica que para apoyar el análisis de flujos de conocimiento primero se requiere modelar el proceso de manera global y detallada. Para el modelado global, se sugiere utilizar una adaptación de la técnica de gráfica rica adaptada (Flores Ríos, Gastélum Ramírez & Rodríguez-Elías, 2010), la cual apoya la identificación del conocimiento requerido y generado durante las actividades, así como las fuentes de conocimiento donde éste es obtenido o almacenado (Flores-Ríos *et al.*, 2014).

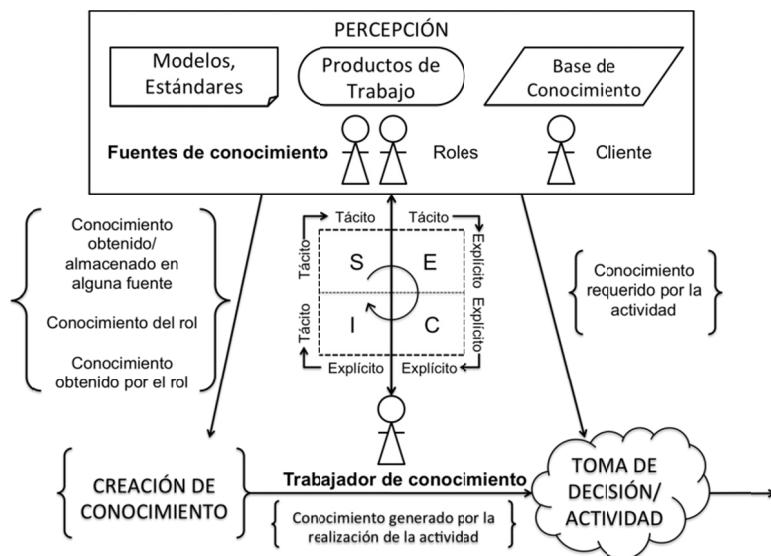


Figura 3 – Modelado global de flujos de conocimiento con entidades proceso software

La gráfica rica adaptada de la Figura 3 sugiere que los procesos de creación de conocimiento del modelo SECI y los elementos del ciclo del conocimiento no son procesos aislados uno del otro, sino que se encuentran interrelacionados debido a que la aplicación del conocimiento permite también incrementar el ya existente (Flores Ríos, Gastélum Ramírez & Rodríguez-Elías, 2010). En dicha figura también aparece el trabajador de conocimiento quien crea, utiliza y transmite tanto conocimiento explícito como tácito. Los roles pueden apoyarse de diversas fuentes de conocimiento (roles, métodos, técnicas, plantillas, recursos o productos de trabajo), que no sólo le ayuden a resolver problemas inesperados, sino a hacer toma de decisiones correctas o incrementar sistemáticamente el conocimiento.

Para el modelado detallado, se utilizó el modelo base de la simbología SPEM (por sus siglas de Software Process Engineering Metamodel). En la Figura 4, se identifican cada una de las entidades proceso software como fuentes de conocimiento del enfoque de ciclo de conocimiento. Sin dejar de reconocer el valor de cada elemento de la gráfica global, la dinámica del ciclo de conocimiento es integrar los procesos del modelo SECI en la creación de nuevo conocimiento. Las actividades son realizadas por los roles, quienes son responsables del uso, generación y almacenamiento del conocimiento en diversas fuentes y tomando decisiones dependiendo del nivel de madurez de la VSE. Por ejemplo, en un proyecto SPI los roles establecen, actualizan, corrigen, verifican y/o validan procedimientos, métodos, productos de trabajo o plantillas, según lo requerido por los atributos de procesos del nivel de capacidad requerido (Flores-Ríos, Rodríguez-Elías & Pino, 2013).

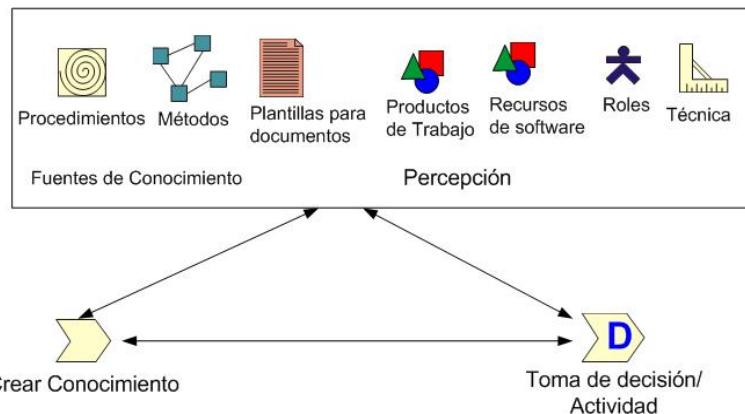


Figura 4 – Modelado detallado de entidades proceso software en el enfoque de ciclo de conocimiento

En este nivel de modelado se establece la relación de las entidades procesos software con la especificación descriptiva y explícita de las actividades definidas en un modelo de referencia de procesos. Por lo que se considera la premisa de que una VSE va adquiriendo madurez en función del nivel de conocimiento y habilidades que poseen los roles para implementar, gestionar, establecer y controlar una serie interrelacionada de procesos definidos en un modelo de referencia, pero también se apoyan de su percepción e interpretación de lo que está sucediendo en el entorno. Esto le permite al

trabajador del conocimiento o equipo de trabajo identificar y especificar acciones de mejora apoyados de nuevas fuentes y tipos de conocimiento.

3.3. Identificar fuentes y tipos de conocimiento.

Para esta fase, se utilizó el metamodelo de conceptos de conocimiento modelado en SPEM-KF (Rodríguez-Elías & Martínez García, 2011). La Figura 5 permite observar cómo se relacionan los tipos o temas de conocimiento (*KTopic*) y fuentes de conocimiento (*KSource*). *Ktopic* es usado para representar temas de conocimiento requeridos o generados por las actividades, así como aquel conocimiento que es almacenado u obtenido de las fuentes de conocimiento. *KSource* se utiliza para clasificar las categorías de fuentes (*KsourceCategory*) y sus tipos (*KSourceType*). Esto permitió detectar que para representar un concepto de conocimiento específico (*Kconcept*) o conjunto de conocimiento (*GroupedKnowledge*) se utiliza la exteriorización del conocimiento requerido, identificado como un tipo de producto de trabajo (*WorkProductType*) (Flores-Rios, Rodríguez-Elías & Pino, 2013).

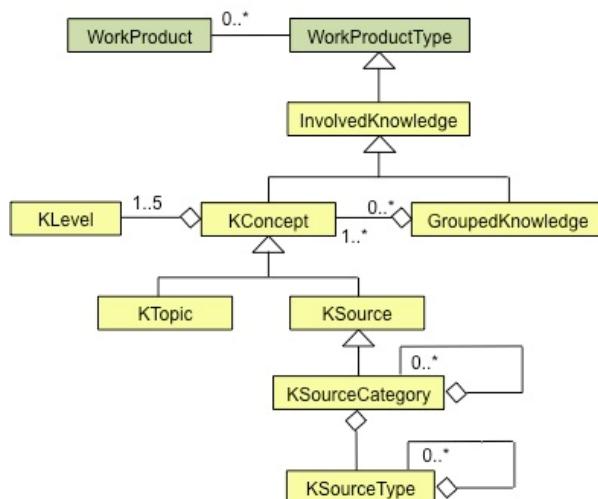


Figura 5 – Metamodelo de conceptos de conocimiento

Desde el inicio de un proyecto SPI, se podrán identificar y capturar los activos de conocimiento por medio de un mapa de conocimiento o plantilla (Flores-Rios *et al.*, 2014b). Esta situación ayuda al equipo de trabajo a detectar claramente los conceptos, fuentes y tipos de conocimiento, así como el nivel de conocimiento (*KLevel*) que poseen los roles para desempeñar las actividades requeridas para un determinado nivel de capacidad de procesos.

3.4. Identificar flujos de conocimiento.

Posteriormente, se identifica el flujo de conocimiento que es aplicado o requerido durante la realización de las actividades o toma de decisiones y las fuentes donde el conocimiento es almacenado u obtenido (Flores-Rios *et al.*, 2014). La Figura 6 muestra

por medio de un diagrama de transferencia de conocimiento (Rodríguez-Elías & Martínez García, 2011) las fuentes de conocimiento, la toma de decisión que realizan los roles y la interacción que tiene cada entidad proceso software con el proceso de creación del conocimiento (modelo SECI) y el modelo integrado.

A continuación, se describen los flujos de conocimiento detectados en dichas entidades:

Socialización (S): Este proceso se presenta cuando los roles interactúan y comparten modelos mentales y habilidades técnicas de acuerdo a su nivel de conocimiento (*KLevel*). Las fuentes de conocimiento (*KSource*) son los roles debido a que comunican e intercambian conocimiento tácito. El colaborar con personas con experiencia y conocimiento especializado según su dominio en métodos y técnicas y/o uso de ciertas plantillas les permitirá a los roles desempeñar las actividades dependiendo de la interacción que tengan con ellos. Algunas veces los roles incrementan su conocimiento tácito cuando otros roles les interpretan el llenado de algunas plantillas de documentos o les mencionan aspectos sobre los productos de trabajo que utilizan en sus actividades. Este escenario se puede ejemplificar cuando una VSE dedicada al desarrollo de software cuenta con personal con poca experiencia y/o posee procesos inestables o no documentados, solicitando la intervención de consultores expertos para que brinden ayuda en la identificación de causas o circunstancias que originan problemas en el desempeño de las actividades (Flores-Rios, Rodríguez-Elías & Pino, 2013). Cuando los roles socializan pueden utilizar herramientas de software que permiten el intercambio de conocimiento, experiencia, confianza e información en ciertas actividades específicas, tales como los portales de conocimiento, sistemas de tele/videoconferencias, mensajería instantánea, entre otros. Por otro lado, los roles se apoyan de tecnología como los videoteléfonos, aparatos de videoconferencias, tableros electrónicos o tecnología groupware.

Exteriorización (E): Por medio de la conceptualización, extracción y articulación en colaboración o reflexión colectiva entre los roles, una parte del conocimiento tácito se convierte en explícito. Para hacer más efectivo este proceso se utilizan herramientas de software del tipo de *newsgroups*, *groupware*, Wikis, flujo de trabajo o sistemas colaborativos; y tecnología para el modelado o generación de gráficos o tecnología audiovisual. Una vez que el conocimiento tácito es convertido en explícito, los roles con escenarios similares pueden encontrar buenas prácticas o soluciones al consultar la plantilla para la captura de activos de conocimiento (Flores-Rios *et al.*, 2014b). Es por esto, que algunas técnicas recomendadas para la exteriorización del conocimiento en proyectos SPI son el transmitir lecciones aprendidas o buenas prácticas utilizando historias, analogías o metáforas para determinadas situaciones o eventos.

Combinación (C): La combinación se refiere al proceso de sistematización de conceptos en el que el rol se dirige a repositorios (bases de conocimiento, repositorios de proyectos, configuración de software, entre otros) para satisfacer sus demandas de conocimiento, generar y modificar conocimiento explícito a partir de otro explícito. Este proceso implica el uso de plantillas, formatos de reportes, recursos de software y hardware para administrar y buscar colecciones de conocimiento bien establecido, almacenado y persistente para generar más conocimiento explícito. Los sitios Web o intranets utilizan índices de búsqueda, agentes inteligentes y/o mapas de conocimiento para localizar ciertos documentos o textos en un dominio específico.

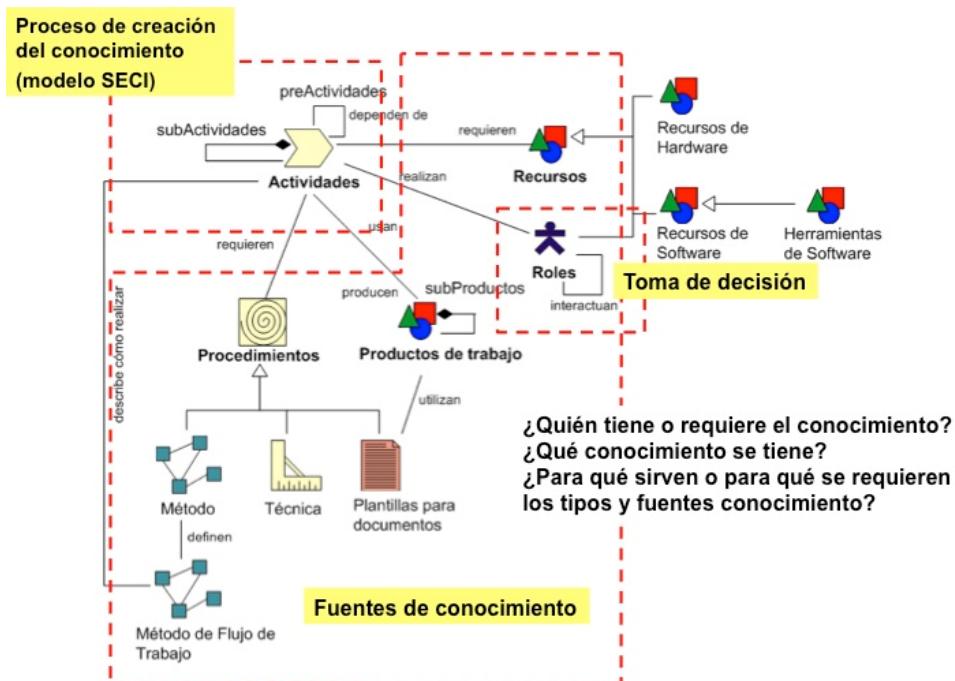


Figura 6 – Análisis de flujos de conocimiento en las entidades proceso software involucradas en un proyecto SPI

Interiorización (I): Cuando se aprende y socializa, al hacer reiteradamente una actividad, el conocimiento explícito se convierte en tácito. De esta forma, si se tienen equipos de desarrollo de software multifuncionales o autodirigidos, éstos aprenden e interiorizan al implementar nuevos procesos, métodos o estrategias como parte de su cultura organizacional. Los productos de trabajo facilitan la transferencia de conocimiento explícito a otros roles, permitiéndoles experimentar indirectamente las vivencias de otros. Aunque la mayoría de las VSE reconoce la importancia de la interiorización del conocimiento, el establecimiento de estrategias para generar nuevo conocimiento tácito se les dificulta y el proceso se les hace complejo. Por otro lado, se han detectado herramientas que apoyan los procesos de socialización, exteriorización y combinación por lo que existe una carencia de recursos de software que faciliten el proceso de interiorización.

Se recomienda que una VSE considere el valor de los flujos de conocimiento en función de los procesos de creación y transferencia de conocimiento como parte de su cultura e implementación de estrategias de GC. Así mismo, se deben de tomar en cuenta las valoraciones y percepción de los roles para medir el impacto del conocimiento y su flujo, en los procesos de ciclo de vida de GC.

El modelo de la Figura 6 se ha diseñado de forma tal que facilite la interpretación de los flujos de conocimiento y sus interacciones en las entidades procesos software. Para conocer la situación actual de una VSE con respecto a su flujo de conocimiento es necesario hacer una auditoria o diagnóstico de conocimiento donde se identifiquen las

fuentes y tipos de conocimiento existentes para detectar qué tipo de conocimiento está fluyendo y cuáles problemáticas se presentan. Por tal motivo, se definió de manera genérica y flexible un marco declarativo de ciclo de vida de GC para ser implantado dentro del proyecto SPI o procesos de una VSE fomentando la aplicación, transferencia y evolución del conocimiento para la generación de productos de software de calidad (Flores-Ríos *et al.*, 2014).

4. Aportaciones e implicaciones prácticas

Los elementos descritos en las secciones anteriores se tomaron en cuenta para la realización de dos trabajos relacionados con SPI: (i) proyecto para obtener el nivel 2 de madurez de una empresa desarrolladora de software siguiendo las normas mexicanas relacionadas con SPI, y (ii) la creación de un método para apoyar la formalización de la estimación de proyectos software en pequeñas organizaciones dedicadas al desarrollo de software. En las siguientes subsecciones se presenta de manera general estos dos trabajos.

4.1 SPI en empresa mexicana

La importancia de la GC es asegurar que el conocimiento de la organización se aplique de manera productiva para su beneficio (Rodríguez-Elías & Martínez García, 2011). En este sentido, los elementos descritos en esta investigación se utilizaron durante el desarrollo de un proyecto SPI en una VSE mexicana de desarrollo de software. El objetivo de este proyecto para la empresa era obtener un nivel 2 de madurez conforme a las normas mexicanas para la implantación de procesos software (NMX-I-059-NYCE-2011) y evaluación de la capacidad de procesos (NMX-I-15504-NYCE-2010). La NMX-I-059-NYCE-2011 establece criterios para que las VSE mexicanas, a través de su adopción e implantación como un modelo de referencia de procesos, puedan ofrecer servicios y productos que alcancen niveles internacionales de calidad (Oktaba & Piattini, 2008). Dicha norma especifica que los niveles de capacidad de procesos y sus atributos se ubican en una escala de cinco niveles. El nivel 2 de capacidad de procesos implica que los 9 procesos requeridos deben de estar planeados, institucionalizados, supervisados y adaptados para cumplir con los objetivos de los procesos y producir los productos de trabajo que se identifican, documentan y controlan adecuadamente como conocimiento explícito (Flores-Ríos, Rodríguez-Elías & Pino, 2013). Es importante resaltar que el objetivo establecido se logró por medio de la ejecución de tres ciclos de mejora en los cuales se utilizaron, entre otros aspectos, los elementos de GC descritos en este artículo.

La metodología de implementación del proyecto junto a los enfoques de flujo de conocimiento permitieron identificar los tipos y fuentes de conocimiento, y verificar empíricamente, si una VSE que desee implementar un proyecto SPI basado en conocimiento requerirá contar con diversos tipos y activos de conocimiento almacenados en la base de conocimiento e identificar el conocimiento que poseen los roles relacionados con los ciclos del proyecto. En este sentido, se analizaron los atributos de procesos, descritos en el modelo NMX-I-15504-NYCE-2010, permitiendo caracterizar los tipos y flujos de conocimiento involucrados en la evaluación de la capacidad del proceso (Flores-Ríos, Rodríguez-Elías & Pino, 2013; Flores-Ríos *et al.*,

2014) para una organización desarrolladora de software. Mediante este trabajo, se logró identificar los elementos relacionados con la GC que permitieron distinguir los dos primeros niveles de conocimiento asociados al nivel de capacidad de procesos. Al final de cada ciclo del proyecto SPI, para un nivel 2 de capacidad, la VSE ha creado nuevo conocimiento explícito almacenado en alguna fuente y conocimiento tácito obtenido y transmitido entre los roles participantes.

4.2 Método para la formalización de procesos

Por otro lado, el marco conceptual de GC presentado en la sección 3, integra las entidades del proceso software utilizadas en la descripción de procesos de creación y transferencia de conocimiento, bajo una perspectiva de flujos de conocimiento. Este marco conceptual fue utilizado como referente para la construcción de un método que apoya la formalización de procesos software en pequeñas organizaciones debido: (i) a la profundidad de la información que contiene relacionada con los procesos de exteriorización y adquisición del conocimiento, y (ii) a que plantea un proceso para la gestión de ciclo de vida de conocimiento dentro de una organización describiendo pasos claros para la identificación de tipos de conocimiento (Carvajal & Márquez, 2013).

Los autores Carvajal & Márquez (2013) presentan un método bajo una estructura detallada, legible y adaptable que permite guiar la formalización de actividades al interior de una empresa dedicada al desarrollo de software. Este método tiene como propósito ser una herramienta que permita a las VSEs llevar a un plano formal su quehacer diario, es decir, plasmar como activo de proceso la manera de cómo son llevadas a cabo las distintas actividades, utilizando un enfoque desde la adquisición de conocimiento, para extraer el conocimiento tácito presente en las diferentes personas que realizan actividades críticas en la organización (Carvajal & Márquez, 2013). El método fue construido siguiendo el proceso de exteriorización (E) con el objetivo de reducir la dependencia del conocimiento tácito y aumentar la creación de conocimiento explícito.

En la evaluación del método se usó el protocolo de método de investigación de caso de estudio en Ingeniería de software propuesto por los autores (Pino *et al.*, 2014) y se definieron las métricas esfuerzo, idoneidad y correctitud para responder a la pregunta de investigación ¿Es idóneo el método definido para formalizar conocimiento en pequeñas organizaciones? (Carvajal & Márquez, 2013). Los resultados obtenidos evidenciaron que el método para la formalización aplicado a la estimación de proyectos de software constituye una herramienta suficiente e idónea para plasmar el conocimiento tácito involucrado en la actividad de estimación. Además, a partir de las métricas recolectadas se obtuvieron resultados positivos que permiten verificar que las características deseadas para el método son las adecuadas para VSEs (Carvajal & Márquez, 2013).

5. Conclusiones

Debido a la naturaleza del conocimiento es necesario distinguir diversos modelos para su gestión. Se requieren esfuerzos complementarios e independientes para la aplicación de estrategias de GC en escenarios de desarrollo y mantenimiento de

software, SPI, innovación de procesos, entre otras actividades puntuizando la transdisciplinariedad entre GC e Ingeniería de software.

En este documento se observó la aportación de KoFI al identificar el tipo de conocimiento requerido y aplicado en el análisis de flujos de conocimiento de las entidades proceso software, bajo los enfoques del modelo SECI, ciclo de conocimiento y modelo integrado de ciclo de vida de GC. El propósito es brindar un marco de trabajo útil para el equipo de trabajo o trabajador del conocimiento que propicie la percepción del entorno, la creación de nuevo conocimiento y toma de decisión efectiva en un proyecto SPI basado en conocimiento.

El marco de trabajo está integrado por un marco conceptual y un marco declarativo. El marco declarativo define siete procesos de conocimiento, donde cada uno representa el flujo de conocimiento y la interrelación de los mismos con la dimensión del proceso. Los elementos obtenidos de esta investigación han sido utilizados de manera satisfactoria en trabajos de mejora de procesos, lo cual evidencia la utilidad de esta propuesta en el contexto de la industria del software.

Referencias bibliográficas

- Anaya, R., Cechich, A. & Henao, M. (2007). A model to classify knowledge assets of a process oriented development. In *Software process improvement for small, medium enterprises, Techniques and cases studies*.
- Capote, J., Llanten Astaiza, C. J., Pardo Calvache, C. J., González Ramírez, A. de J. y Collazos, C. A. (2008). Gestión del Conocimiento como apoyo para la mejora de procesos software en las micro, pequeñas y medianas empresas, *Revista Ingeniería e Investigación*, 28 (1), 137-145.
- Carlsson, S. A. (2002). Towards an understanding and conceptualization of knowledge managing within the context of inter-organizational networks, Third European Conference on Organizational Knowledge, Learning, and Capabilities (OKLC 2002).
- Carvajal, L. F. & Márquez, C. A. O. (2013). Método que apoye la formalización de la estimación de proyectos software en pequeñas organizaciones. Tesis de licenciatura. Universidad del Cauca. Colombia.
- Choo, C. W. (1999). *The knowing organization: How organizations use information to construct meaning, create knowledge, and make decisions*. Oxford University Press. Oxford, USA.
- Falbo, R. A. y Bertollo, G. (2009). A software process ontology as a common vocabulary about software processes. *International Journal of Business Process Integration and Management*, 4, 239-250. doi: 10.1504/IJBPM.2009.032281
- Flores-Ríos, B. L., Astorga Vargas, M. A., Rodríguez-Elías, O. M., Ibarra-Esquer, J. E. & Andrade, M. D. C. (2014). Interpretación de las Normas Mexicanas para la Implementación de Procesos de Software y Evaluación de la Capacidad bajo un Enfoque de Gestión de Conocimiento. *Revista Facultad de Ingeniería*, 71 (71), 85-100.

- Flores Rios, B. L., Gastélum Ramírez, S. L. & Rodríguez-Elías, O. M. (2010). Modeling Knowledge Flows in Software Projects Management Processes, International Conference on Knowledge Management and Information Sharing (KMIS), España, 213-217.
- Flores-Rios, B. L., Ibarra-Esquer, J. E., Pino, F. J., Astorga-Vargas, M. A., González-Navarro, F. F. & Rodríguez-Elías, O. M. (2014b) La Exteriorización de activos de conocimiento en mejora de procesos software. Encuentro Nacional de Ciencias de la Computación (ENC). México.
- Flores-Rios, B. L., Rodríguez-Elías, O. M. & Pino, F. J. (2013). Administración del producto de trabajo como gestión de conocimiento explícito de conformidad con el estándar ISO/IEC 15504. CONISOFT. México, 29-36.
- Garzón Castrillón, M. A. & Fisher, A. L. (2008). Modelo teórico del aprendizaje organizacional. *Pensamiento y Gestión*, 24, 195-224.
- Gopesh, A., Ward B. & Mohan, T. (2007). Role of explicit and tacit knowledge in six Sigma projects: An empirical examination of differential project success. *Journal of Operations Management*, 8(4), 303-315. doi: 10.1016/j.jom.2009.10.003
- Guo, J. M. & Wang, Y. L. (2008). Context modeling for knowledge flow, Proceedings of 2008 IEEE International Conference on Information Reuse and Integration, IEEE IRI-2008, Institute of Electrical and Electronics Engineers Computer Society, Piscataway, NJ 08855-1331, USA, pp. 330-335.
- Gupta, A. & Govindarajan, V. (2000). Knowledge flows within multinational corporations. *Strategic Management Journal*, 21 (4), 473-496.
- ISO/IEC TR 24774. (2007). Software and systems engineering – Life cycle management – Guidelines for process description.
- Kautz K. & Nielsen, P. A. (2004). Understanding the implementation of software process improvement innovations in software organizations, *Information Systems Journal*, 14 (1), 3-22. doi: 10.1111/j.1365-2575.2004.00156.x
- Kruchten, P. (1998). The Rational Unified Process: An Introduction, Addison Wesley.
- Mathiassen, L. & Pourkomeylian, P. (2003). Managing Knowledge in a Software Organisation. *Journal of Knowledge Management*, 7(2), 63-80.
- Niazi, M., Wilson D. & Zowghi, D. (2005). A framework for assisting the design of effective software process improvement implementation strategies. *Journal of Systems and Software*, 78 (2), 204-222. doi: 10.1016/j.jss.2004.09.001
- Nissen, M. & R. Levitt. (2002). Dynamic models of knowledge flow dynamics. Disponible en: <http://cife.stanford.edu/online.publications/WP076.pdf>.
- NMX-I-059/02-NYCE-2011. (2011). Tecnología de la Información - Software - Modelos de Procesos y Evaluación para el Desarrollo y Mantenimiento de Software. Parte 02, Requisitos de Procesos (MoProSoft). NYCE. México.

- Nonaka, I. & Takeuchi, H. (1995). *The knowledge-creation company: How Japanese companies create the dynamics of innovation.* 1st. Ed. Oxford University Press. USA. 304.
- Oktaba, H. & Piattini, M. (2008). *Software Process Improvement for Small and Medium Enterprises.* IGI Global, USA.
- Pino, F. J., García, F. O., Ruiz, F. & Piattini, M. (2006). Adaptación de las normas ISO/IEC 12207:2002 e ISO/IEC 15504:2003 para la evaluación de la madurez de procesos software en países en desarrollo. *Revista IEEE América Latina*, 4 (2), 17-24.
- Pino, F., García, F., & Piattini, M. (2007). Priorización de procesos como apoyo a la mejora de procesos en pequeñas organizaciones software. In XXXIII Conferencia Latinoamericana de Informática (CLEI 2007). 77.
- Pino, F. J., García, F. & Piattini, M., (2014). A Retrospective of the Use of Action-Research and Case Study in Software Process Improvement. *Informe técnico*, Junio. 34 p.
- Rodríguez-Elías, O. M. & Martínez García, A. I., (2011). *Diseño de sistemas y estrategias de gestión del conocimiento: Un enfoque metodológico orientado a procesos y flujos de conocimiento*, Editorial Académica Española, 236 p.