

# **PROPUESTA DE INTEGRACION DEL PROCESO DE DESARROLLO DE SOFTWARE Y LA GESTION DEL CONOCIMIENTO BASADO EN UNA VISION HOLISTICA**

## **“PROPOSAL FOR INTEGRATION OF SOFTWARE DEVELOPMENT PROCESS AND KNOWLEDGE MANAGEMENT BASED ON A HOLISTIC VISION”**

**Carlos J. Guevara F.**

Universidad Nacional Experimental de Guayana, CITEC, Puerto Ordaz, Venezuela, [carlos.guevaraf@gmail.com](mailto:carlos.guevaraf@gmail.com)

**Aixa Viera**

Universidad Nacional Experimental de Guayana, CITEC, Puerto Ordaz, Venezuela, [avierabl@yahoo.com](mailto:avierabl@yahoo.com)

### **RESUMEN**

La expansión de las Tecnologías de Información y Comunicaciones produjo un acelerado incremento en la generación del conocimiento. El software, elemento importante en el desarrollo de la sociedad del conocimiento, al igual que el conocimiento es intangible, de construcción lógica, creativa y personal. En este trabajo se presenta una propuesta de integración del proceso de desarrollo del software y la gestión del conocimiento basado en una visión holística y en técnicas de la Ingeniería del Software, que facilita abordar los problemas de rápida obsolescencia del software y del conocimiento; así como el proceso de diseño de herramientas para su gestión. Se realizó una investigación documental y aplicada, utilizando análisis de contenido y su correspondiente categorización, lo que permitió generar un modelo para construir un sistema de información de gestión del proceso de generación de conocimiento, con el cual se monitorea la generación de competencias investigativas de un grupo de estudiantes de maestría. Se concluye que existe una relación dialéctica entre la Gestión de Conocimiento y los procesos de elaboración de software, relación que se explica mediante la expresión “circularidad concéntrica” con la cual denominamos la interrelación existente entre ambos, pues no se sabe dónde termina una y empieza la otra.

**Palabras claves:** Software, Conocimiento, Ingeniería de Software, Gestión de Conocimiento

### **ABSTRACT**

The expansion of Information and Communications Technologies produced a rapid increase in the generation of knowledge. The software, important issue in the development of the knowledge society, as knowledge is intangible, logical construction, creative and personal. This paper presents a proposal for integrating software development process and knowledge management based on a holistic and techniques of software engineering that facilitates addressing the problems of rapid obsolescence of software and knowledge, as well as design process management tools. We performed a documentary and applied research, using content analysis and the corresponding categorization, allowing to create a model for building a information system to management the knowledge generation process, which monitors the generation of investigative skills of a group of graduate students. We conclude that there is a dialectical relationship between knowledge management and software development processes, a relationship that is explained by the term "concentric circular" with which we have called the interrelationship between them, it is not known where one ends and the other begins.

**Keywords:** Software, Knowledge, Software Engineering, Knowledge Management

# 1. INTRODUCCIÓN

Desde la antigüedad hasta aproximadamente el siglo XVII el factor estratégico de desarrollo del mundo era la tierra por lo que este período se le denomina la era agrícola, con la aparición de la máquina hacia el siglo XVIII se da inicio de la primera revolución industrial, período éste que se extiende hasta inicio del siglo XX, a partir del cual las tecnologías toman un papel preponderante como principal factor estratégico de desarrollo, por lo que varios autores, considerando la influencia de la tecnología, han denominado al siglo XX como la era de la Segunda Revolución industrial. Toffler A. (1995), considera que la economía tiene ciclos de vida, que cada ciclo es mucho más corto que el anterior y que el hombre recorre cada nueva etapa 5 veces más rápido que la anterior. Esta evolución se puede visualizar como se presenta en la siguiente gráfica:

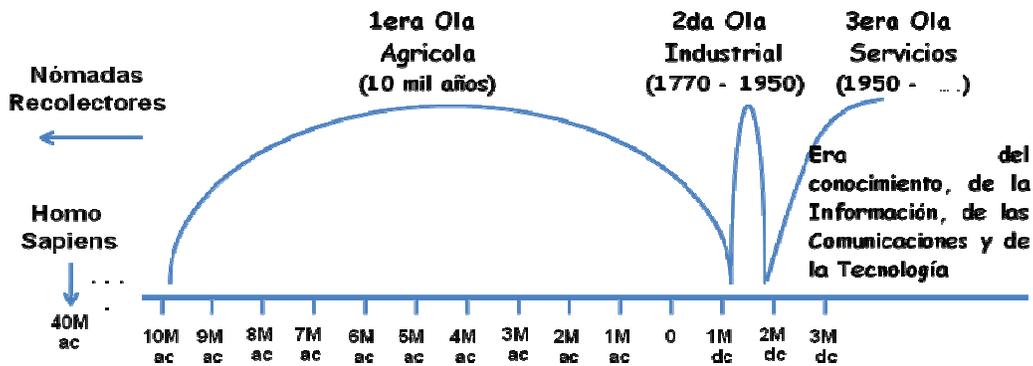


Gráfico 1. Olas de de la Humanidad (Toffler, 1995)

La denominada por Toffler como la tercera ola de la humanidad o la ola de servicios, coincide con la aparición de la primera computadora comercial (UNIVAC I) o la segunda generación de computadoras que causó gran impacto a nivel mundial en cuanto a la forma de tratar y procesar información, es por ello que en el continuo avance de la humanidad hace que esta tendencia apunte a considerar, en el siglo XXI, al conocimiento como el factor de mayor relevancia de las organizaciones, convirtiéndose en el elemento de mayor valor estratégico de desarrollo.

Basado en información compilada por los autores de este trabajo, de las siguientes fuentes: (Grifantini, 2009; intel, 2010; García, 2010), a partir de la aparición de los circuitos integrados en la década de los 70, y tomando como muestra los microprocesadores comerciales de hoy en día, se puede evidenciar el rápido desarrollo que ha tenido la tecnología en el ámbito de las computadoras, tal como se muestra en el gráfico siguiente, señalándose la cantidad de transistores que se han incorporado en éstos circuitos integrados

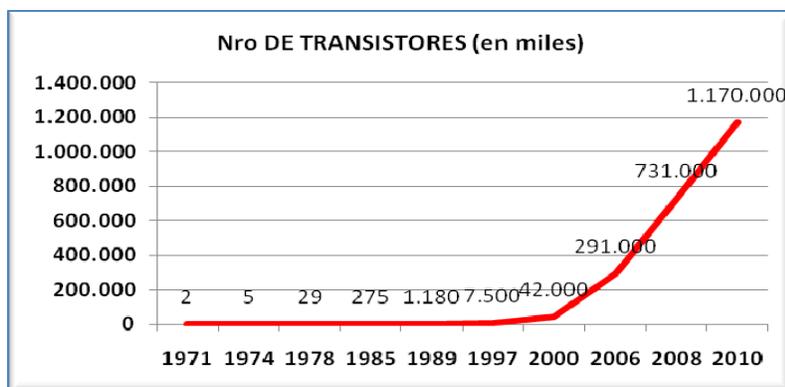
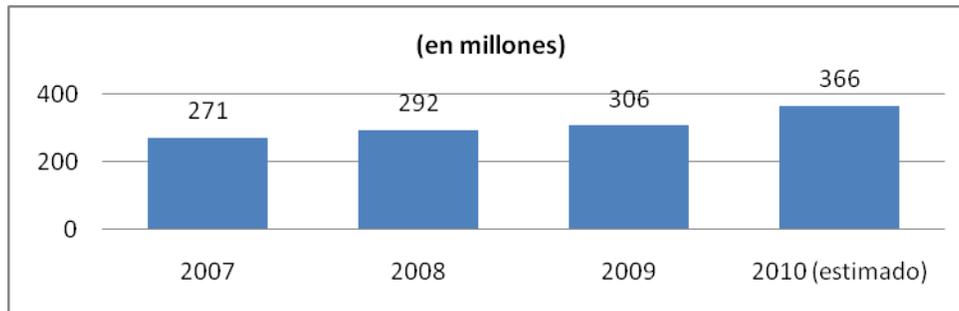


Gráfico 2. Evolución de cantidad de Transistores en Microprocesadores

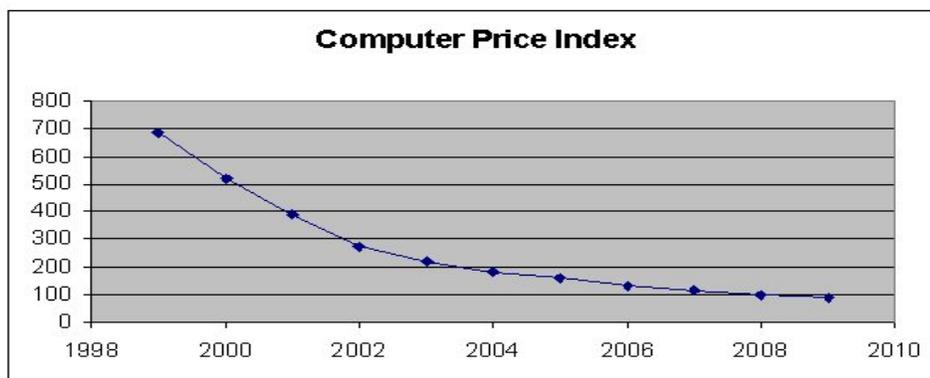
	PROCESADOR	Nro de transistores (miles)
1971	Intel 4004	2
1979	intel 8080	5
1984	intel 8086	29
1985	intel 386	270
1989	Intel 80486, RISC I860	1.000
1997	Pentium MMX	4.500
2000	Pentium IV	42.000
2006	Intel Core Duo 64 Bits	291.000
2008	Intel Core i7	731.000
2010	Seis núcleos Core i7	1.170.000

Si se considera el número de PC existentes en el mundo también se demuestra un acelerado avance, según Gartner (2010), para 2008 se situaba en más de 1.000 millones de unidades, con una tasa de crecimiento promedio en los últimos años de un 12% y estima que para 2014 habrá en uso más de 2.000 millones de PC. En el caso de las ventas de PC en los últimos años se muestra en el siguiente gráfico



**Gráfico 3. Venta de PC a nivel mundial en los últimos años (Gartner, 2010)**

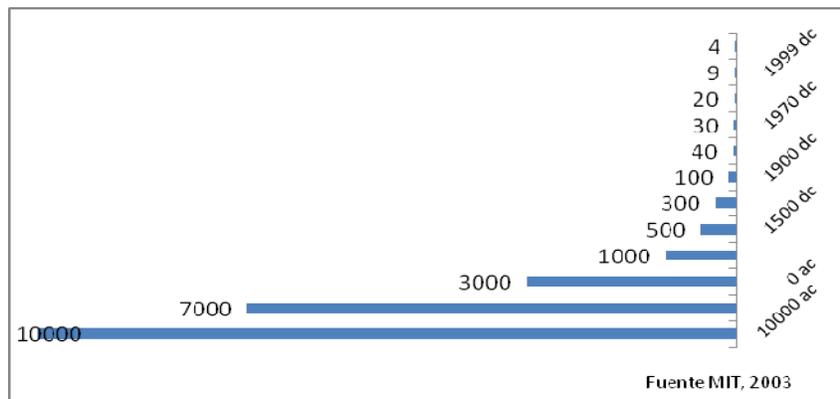
Adicionalmente, el precio de las PC ha decrecido a una tasa del 20% anual en el período 1998 – 2003, y en un 11% en el período 2003 - 2009 como se observa en el siguiente gráfico (tomado de <http://www.freeby50.com/2009/04/cost-of-computers-over-time.html>) :



**Gráfico 4. Evolución de Precios de las Computadoras. (The Bureau of Labor Statistics, 2009)**

Aunado a esto está el desarrollo de internet que ha propiciado el intercambio de información entre los distintos usuarios de la red, facilitando la transferencia y globalización del conocimiento a nivel mundial

En un estudio que realizó el Massachusetts Institute of Technology (MIT), tomado de Pimentel (2005), presenta el tiempo que tarda la humanidad en duplicar su conocimiento, como se aprecia en el gráfico 5 que se muestra a continuación, este tiempo se reduce de forma exponencial, comienza desde el neolítico hace 10.000 años ac y en esta primera etapa tardó 7000 años en duplicar su conocimiento, para el 2003 solo tardó 4 años y se estima que para el 2020 el conocimiento se duplicará cada 78 días



**Gráfico 5. Etapas de duplicación del conocimiento humano**

Con lo antes expuesto se puede afirmar que la introducción de las computadoras y la penetración de las Tecnologías de Información y Comunicaciones (TIC's) en casi todas las actividades que ejecuta la humanidad ha traído como consecuencia un acelerado incremento en la generación del conocimiento, convirtiendo a éste en el principal activo de las organizaciones y la capacidad de aprendizaje en el factor estratégico de desarrollo, por lo que muchos autores no dudan en decir una estamos en presencia de la era del Conocimiento.

Uno de los factores que ha contribuido en el desarrollo de la sociedad del conocimiento, junto al impresionante avance del hardware y de las comunicaciones, es el software, sin embargo entre los problemas que éste tiene que enfrentar es su rápida obsolescencia. Por consiguiente es necesario realizar revisiones continuas a los procesos y paradigmas de desarrollo de software, dado que deben ir evolucionando para adaptarse a las nuevas exigencias de las aplicaciones y usuarios actuales. Al ser el conocimiento y el software productos eminentemente intangibles dependientes de procesos de construcción lógicos, creativos y personales, nos coloca también ante otro problema de diseñar herramientas que faciliten su gestión.

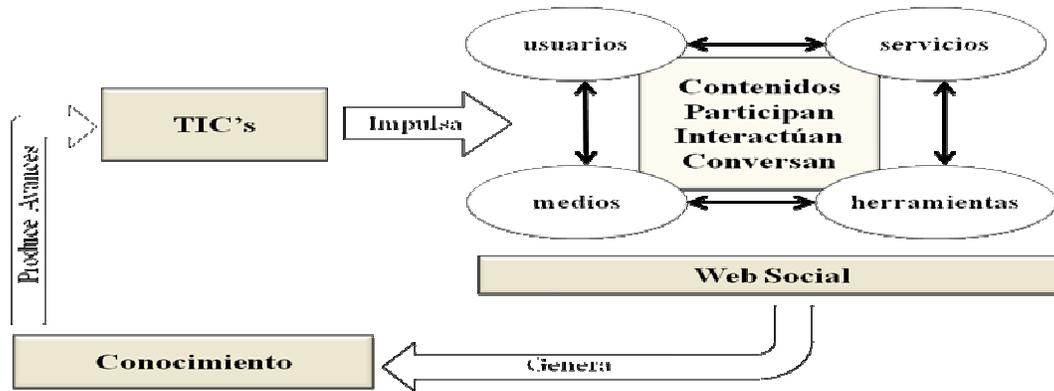
## 2. SOFTWARE Y CONOCIMIENTO

El software es un producto abstracto e intangible (Sommerville, 2005) y la programación requiere de habilidad e ingenio (Knuth, 1974). El proceso de elaboración de software, considerado como el conjunto de actividades necesarias para la construcción del software requiere, conocimientos de la ciencia de la computación, de los métodos y herramientas específicos de la ingeniería, así como del intelecto y la creatividad de las personas. Así mismo, es un proceso lógico, creativo e intelectual, por tal motivo depende de las decisiones y juicios que toman las personas involucradas en él. Al ser un proceso intelectual, inevitablemente necesita de conocimiento y éste es un proceso intrínsecamente relacionado a las personas, en consecuencia el desarrollo de software es de naturaleza cognitiva por lo que su gestión toma una relevante importancia dentro de la Ingeniería del Software.

En la actualidad no hay un consenso sobre la definición de conocimiento, sin embargo basado en su origen etimológico y en las definiciones de autores como Nonaka y Takeuchi (1995), Grant, (1996), Davenport y Prusak (1998), se puede entender como el conjunto de información de los objetos, tomada y almacenada por el sujeto mediante la experiencia o aprendizaje o por la introspección, y la interrelación que se da entre ellos para entender lo que son y no son (Guevara, 2009<sup>1</sup>), por lo que el conocimiento es un proceso en el que intervienen las personas con una base de información previa, influenciadas con sus experiencias propias, sus expectativas futuras y el ambiente en el que se desempeñan, lo que les permite establecer relaciones lógicas e interpretaciones que les ayuda a tomar decisiones y generar nuevos conocimientos. El conocimiento, al igual que el software, depende de las personas, es abstracto, es lógico no físico, se construye y no se fabrica, no se consume solo se desactualiza.

<sup>1</sup> Ponencia presentada en un Congreso de Estudiantes de Informática en Tarapoto, Perú, no publicada.

Como se mencionó anteriormente, el desarrollo de las TIC's ha permitido la evolución del software tanto en complejidad como en el alcance y éste se involucra en muchas de las actividades que realiza el hombre en la sociedad actual. Las nuevas herramientas informáticas permiten la socialización y transferencia de conocimiento facilitando la formación de redes sociales constituyendo, lo que muchos autores coinciden en denominar la Web Social, por tal razón los autores de esta ponencia proponen el Modelo de Generación de Conocimiento basado en las TIC's como se presenta en la siguiente figura



**Figura 1. Modelo de Generación de Conocimiento basado en las TIC's**

Este modelo lo que propone es una relación concéntrica y en espiral entre el Conocimiento, las Tecnologías de la Información y Comunicaciones (TIC's) y la Web Social. De manera resumida lo que plantea es que el desarrollo del Conocimiento en diferentes áreas, tanto en las ciencias básicas como en las empíricas ha producido avances extraordinarios en las TIC's y ésta a su vez ha permitido la evolución de lo que en principio se concibió como una web de computadoras hacia una Web Social. En la Web Social las personas, mediante el uso de herramientas informáticas, medios de comunicación (principalmente internet) y servicios web, interactúan entre sí, participan, intercambian e incluso discuten la generación de nuevos contenidos. Esta característica de la Web Social permite la socialización y externalización del conocimiento, propicia la creación de grupos de interés, que pueden estar distribuidos geográficamente en sitios muy distantes y pertenecer a distintas disciplinas del saber, lo facilita la combinación y generación de nuevos conocimientos. Estos nuevos conocimientos producen nuevos avances en las TIC's por lo que se repite el ciclo descrito anteriormente pero en un nuevo estadio de avance tecnológico y de conocimiento, lo que nos ubica en un proceso en espiral en donde en cada ciclo nos colocamos en una capa superior.

Uno de los elementos claves en el modelo descrito anteriormente es la producción de nuevos programas (software) que faciliten el proceso implícito en él, por lo que debemos aplicar técnicas para su desarrollo que se ajusten a las exigencias actuales de los programas que cada día se hacen más complejos pero que requieren tiempos de desarrollo cada vez más rápidos.

### 3. INGENIERÍA DE SOFTWARE Y GESTIÓN DE CONOCIMIENTO

La Ingeniería de Software no es nueva, su origen se remonta a finales del año 1968 cuando su noción fue propuesta en una conferencia celebrada en Garmisch-Alemania, organizada por el Comité de Ciencia de la OTAN para tratar lo que en su momento se denominó "la crisis del software", producto de la forma indisciplinada en que se estaba desarrollando el software para la época y los altos costos que se estaban generando. La ingeniería de software es el establecimiento y uso de principios sólidos de la ingeniería para obtener económicamente un software confiable y que funcione de modo eficiente en máquinas reales (Bauer, 1968), es el estudio de los principios y metodologías para desarrollo y mantenimiento de sistemas de software (Zelkowitz, 1978), es la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento del

software: es decir, la aplicación de ingeniería al software (IEEE, 1990), es una disciplina de la ingeniería cuya meta es el desarrollo costeable de sistemas de software (Sommerville, 2005), integra el proceso, los métodos, y las herramientas para el desarrollo de software de computadora (Pressman, 2006). En consecuencia, la Ingeniería de Software es una disciplina que aplica elementos de la ingeniería, utilizando teorías, métodos y herramientas, para el desarrollo de software de una manera sistemática y disciplinada.

De acuerdo a Pressman (2006), la Ingeniería de Software está constituida básicamente por 4 capas: calidad, métodos, proceso y herramientas, y sostiene que “el fundamento de la ingeniería del software es la capa de proceso”. De igual manera (Sommerville, 2005) afirma “Las nociones fundamentales de procesos y la organización del sistema son la base de todas éstas técnicas y éstas son la esencia de la ingeniería del software”, por lo que podemos considerar que lo primordial en la Ingeniería de Software es la capa de proceso, es decir, el conjunto de actividades requeridas para desarrollar un software.

Ahora bien, por una parte tenemos que el proceso de desarrollo del software hace uso intensivo del conocimiento y tiene una fuerte dependencia de las personas y por la otra a la Ingeniería de Software que provee teorías, métodos y herramientas necesarias para que este proceso de desarrollo sea sistemático y disciplinado, pero la Ingeniería de Software requiere a su vez conocimientos de diferentes índoles y su gestión es necesaria por la naturaleza misma del proceso. Debido a esto último, y en el camino de las indispensables construcciones modélicas, en la figura 2 se presenta una visión holística del proceso de desarrollo de Software

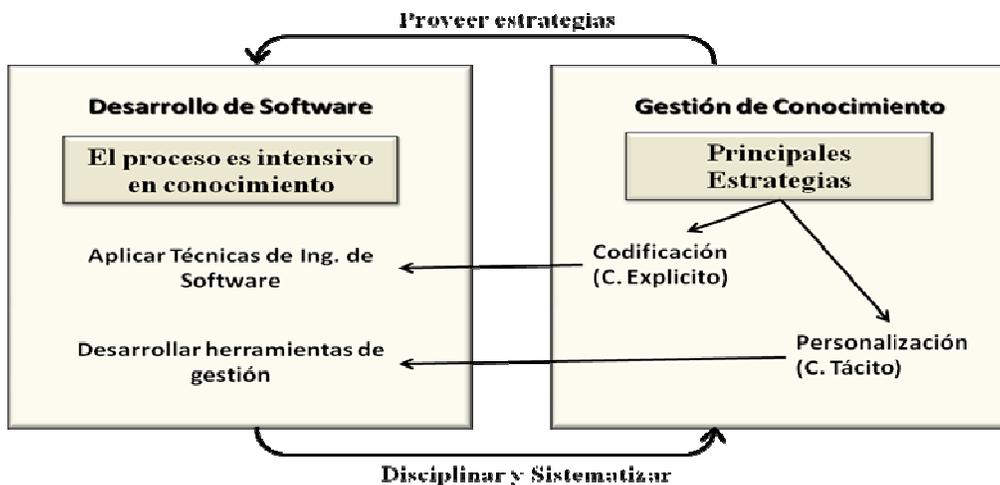


**Figura 2. Proceso de Desarrollo de Software. Visión Holística**

En el Modelo que se presenta se quiso ilustrar que en el proceso del desarrollo del software, el conocimiento se encuentra en diferentes dominios y está distribuido entre los desarrolladores, los usuarios y el mundo externo. Los objetivos se construyen de forma dinámica. El desarrollo de los programas es un proceso de construcción lógico, creativo que reside en la mente humana en donde intervienen diferentes personas, que pueden estar distribuidos en tiempo y espacio. La organización provee herramientas y estrategias que permiten su retroalimentación para preservar el conocimiento obtenido y así facilitar su mejora continua, considera dos estrategias principales en la Gestión de Conocimiento, la Personalización para el manejo del conocimiento tácito y la Codificación para tratar el Conocimiento Explícito

#### 4. RESULTADOS

En el 2010 se aplicó este modelo para construir el prototipo de una herramienta automatizada para gestionar el proceso de generación de conocimiento durante el desarrollo de competencias investigativas de un grupo de estudiantes de la maestría de Tecnología de la Información que se imparte en la Universidad Nacional Experimental de Guayana, para elaborar su anteproyecto de grado. Esto nos permitió integrar técnicas de ingeniería de software con la gestión de conocimiento. El resultado se puede presentar en la siguiente figura:



**Figura 3. Integración Desarrollo de Software y Gestión de Conocimiento**

De acuerdo a Kant (1973), la mente humana es un participante activo y formativo de lo que ella conoce. Si concebimos el desarrollo del software como un proceso intensivo en conocimiento, el sujeto (desarrollador del software) que en él interviene tiene un cúmulo de conocimientos y experiencias que necesariamente lo lleva a interpretar al objeto (gestión del conocimiento) desde su realidad aplicando de manera disciplinada, técnicas de Ingeniería de Software al conocimiento explícito pero con carácter creativo, esto condujo poder construir un software bajo la experiencia y expectativas de los participantes en el desarrollo de competencias investigativas. Por otra parte, este conocimiento creativo y personal (tácito) del sujeto, hoy en día se busca gestionarlo para hacerlo explícito, por lo que se apoya en el software para sistematizarlo. En nuestro caso la herramienta construida (software) facilitó gestionar el conocimiento adquirido en el desarrollo de competencias investigativas, retroalimentando a los sujetos en cada una de las fases de manera reiterativa lo que permitió alcanzar las metas establecidas, pero esto a su vez produjo el desarrollo de nuevas relaciones que permitió generar conocimientos y habilidades que conllevó a incorporar otras estrategias en el desarrollo del software. Lo aquí planteado nos pone en presencia de un enfoque dialéctico, en donde el sujeto conocedor (desarrollador de software) y el objeto (gestión del conocimiento) están en continua interacción, éstos no son inmutables sino que están en perenne transformación y evolución.

Al final se concluye que existe una relación dialéctica entre la Gestión de Conocimiento y el proceso de elaboración del software que se efectúan dentro de parámetros de la Ingeniería del Software, relación que se explica mediante la expresión de “circularidad concéntrica” con la cual hemos querido denominar la interrelación que existe entre ambos, pues no se sabe dónde termina una y empieza la otra. Por una parte la Gestión de Conocimiento provee estrategias que permitan convertir el conocimiento tácito en explícito, incluso el que reside en el mismo proceso de construcción del software, para así crear herramientas que faciliten su aplicación y por la otra el Desarrollo de Software permite crear herramientas que permiten sistematizar la Gestión del Conocimiento, aplicando técnicas de Ingeniería de Software. En otras palabras, estamos en una estructura dinámica o sistema en donde sus componentes están estrechamente relacionados por lo que debemos tener un enfoque holístico e interdisciplinario de los elementos que los constituyen

## REFERENCIA

- Bauer, F. (1968). "Ingeniería del Software". *Informe Primera Conferencia sobre desarrollo de Software*. Comité Científico de la OTAN, Garmisch, Alemania, 7 a 11 octubre 1968.
- Davenport, T.; Prusak, L. (1998), "Working Knowledge: How Organizations Manage What They Know", Harvard Business School Press.

- Garcia, O. (2010). "Intel Core i7-980, con seis núcleos, ya a la venta", *Revista PC WORLDdigital*, <http://www.idg.es/pcworld/Intel-Core-i7-980-con-seis-nucleos-ya-a-la-venta/doc90860.htm>. 01/03/2010. (15/10/2010).
- Gartner (2010). Pronóstico de Garner. Diarioti.com, <http://www.diarioti.com/gate/n.php?id=26028> , 05/03/2010 (10/09/2010)
- Grant RM (1996). "Toward a knowledge-based theory of the firm". *Strat. Manag. J.* Vol 17, pp 109-122.
- Grifantini, K. (2009). "The computer chip has evolved from a simple integrated circuit to a microprocessor with millions of transistors". *Revista Technology Review, MIT*. January/February 2009. <http://www.technologyreview.com/article/21886/>, (19/10/2010)
- IEEE Computer Society (1990). IEEE SA 610.12-1990 - IEEE Standard Glossary of Software Engineering Terminology (2/2/2011).
- Intel Corporation (2010). Procesador Intel Core i7 Extreme Edition. [http://www.intel.com/es\\_Es/Consumer/Products/Processors/corei7-extreme.htm](http://www.intel.com/es_Es/Consumer/Products/Processors/corei7-extreme.htm) (12/11/2010)
- Kant E. (1973, orig. 1787). *Crítica de la razón pura*. Buenos Aires: Losada.
- Knuth, D. (1974). "Computer programming as an art". *CACM*, Vol 17, Nro 12, pp 667-673. Diciembre, 1974 <http://cacm.acm.org/magazines/1974/12>. (01/07/2010)
- Nonaka I, Takeuchi H (1995). *The Knowledge-Creating Company*. Oxford University Press. Nueva York, NY, EEUU
- Pimentel, D (2005). Introducción a los sistemas complejos de diseño de información. Universidad de Buenos Aires. [http://www.simpleweb.ws/fadu/teorica\\_fadu\\_dg3.pdf](http://www.simpleweb.ws/fadu/teorica_fadu_dg3.pdf) (01/08/2009)
- Pressman, R. (2006). *Ingeniería del Software: Un enfoque práctico*. Sexta edición. Mc Graw Hill. México. DF,
- Sommerville, I. (2005). *Ingeniera de Software*. Séptima Edición. Pearson Educación, S.A. Madrid
- The Bureau of Labor Statistics (2009). Índice de Precios al Consumidor. United State Department of Labor <http://www.bls.gov/cpi/#tables> (08/09/2010)
- Toffler A. (1995). *La Tercera Ola*. Edición 11. Editor Plaza & Janes.
- Zelkovitz, M. (1978). "Perspectives on Software Engineering", *ACM Computing Surveys*. Vol 10, Nro 2, Junio 1978, pp 197-216

### ***Autorización y Renuncia***

*Los autores autorizan a LACCEI para publicar el escrito en las memorias de la conferencia. LACCEI o los editores no son responsables ni por el contenido ni por las implicaciones de lo que esta expresado en el escrito.*