



IDENTIFICACI N DE LA DIAL CTICA HEGELIANA EN UN PROCESO  GIL DE LA INGENIER A DEL SOFTWARE. ESTUDIO PRELIMINAR

(Dialectical concepts identified in an agile software engineering process.
Preliminary study)

Recibido: 11/06/2011 Aceptado: 14/09/2011

Mari o, Sonia

Universidad Nacional del Nordeste, Argentina
simarinio@yahoo.com

Godoy, Mar a

Universidad Nacional del Nordeste, Argentina
mvgodoy@exa.unne.edu.ar

Alfonzo, Pedro

Universidad Nacional del Nordeste, Argentina
plalfonzo@hotmail.com

RESUMEN

Se presenta un estudio preliminar con el objeto de aportar consideraciones te ricas transdisciplinarias que sustenten la presencia de la teor a dial ctica -entendida desde las Ciencias Cognitivas- en SCRUM, un proceso de gesti n de proyectos de software. El trabajo se compone de las siguientes secciones: en la primera, se resumen generalidades que contextualizan la inform tica y la ingenier a del software en el trabajo. La segunda secci n sintetiza aspectos relevantes del marco te rico a partir del cual se elabora la propuesta. En la tercera secci n se establece una analog a entre la dial ctica desde la perspectiva de Samaja y SCRUM. Finalmente, se exponen las discusiones y las conclusiones.

Palabras claves: Metodolog as, Ciencias Cognitivas, Ingenier a del software, Dial ctica, SCRUM.

ABSTRACT

The aim of this paper is present a preliminary study and considerations linking the dialectical theory -from the Cognitive Sciences- and SCRUM, a process of project management software. The work consists of the following sections. In the first is an overview to contextualize the computer science and software engineering at work. In the second, it sets general aspects that contextualize the theme. The third section establishes an analogy between the dialectic and SCRUM process. Finally, the some discussions and conclusions are presented.

Keywords: Methodology, Cognitive Sciences, Software engineering, Dialectics, SCRUM.



1. INTRODUCCIÓN

1.1. LA DISCIPLINA INFORMÁTICA

Las Ciencias Cognitivas surgidas en los años cincuenta, abordan “estudios sobre aquellos aspectos y procesos de la mente que nos distinguen como seres humanos” (Aliseda, 2007, p.23). Surge del aporte de seis disciplinas científicas: Inteligencia Artificial, Neurociencia, Psicología Cognitiva, Lingüística, Filosofía Cognitiva (o de la mente), y Antropología Cognitiva. Howard Gardner en el año 1985 introduce el Hexágono Cognitivo para describir las interrelaciones de estos seis campos de la ciencia.

En este siglo caracterizado por el paradigma emergente, la transdisciplinariedad puede tratarse desde diversas perspectivas, una es desde las Ciencias Cognitivas. Como se mencionó en Mariño y otros (2010, p.295), “la informática puede considerarse como una transdisciplina, ya que sus métodos y técnicas son aplicables a la resolución de una variabilidad muy extensa de problemas que abarcan diversos campos de actuación”.

En este sentido, y con el propósito de continuar estableciendo vinculaciones iniciales entre ambos campos disciplinares: la Informática y las Ciencias Cognitivas, se hace lugar a consideraciones de distintos autores en referencia a la epistemología de la primera. En este sentido, se retoma lo expuesto por Barchini y otros (2007, p.2), quienes consideran que hoy se está:

“En condiciones de afirmar que la informática, por su génesis y por sus características intrínsecas, es una disciplina científico-tecnológica y, en su interacción con otras disciplinas, es bio-psico-socio-tecno-cultural”. La informática (Red de Universidades Nacionales con Carreras en Informática (UNCI), 2006), denominada como ciencias de la computación por la escuela estadounidense, se compone de nueve disciplinas siendo una de ellas la Ingeniería del Software (IS).

La IS nació a partir de las ingenierías de sistemas y de hardware. Menciona como elementos clave: los métodos, las herramientas y los procedimientos que facilitan el control del proceso de desarrollo de software y brinda a los desarrolladores las bases de calidad de una forma productiva (Pressman, 2005).

Barchini y otros (2007) presentaron una taxonomía referente al campo de la informática y sus subdisciplinas. Se señalan en letras negritas los conceptos abordados en el presente trabajo:

- Objetos: **sistemas**, algoritmos, sistemas inteligentes, web, bases de datos, telecomunicaciones, arquitecturas, etc.
- Fenómenos: **gestión de sistemas**, análisis y diseño de algoritmos, performance, interoperabilidad, etc.
- SScibernética, etc. En Mariño y otros (2010) se incorporó la **Teoría Dialéctica**.



Por otra parte, fueron previamente analizados e incorporados a la propuesta de Barchini y otros (2007) elementos en las categorías “teorías” y “métodos y aplicaciones prácticas” (Mariño y otros, 2010). Se señalan más abajo los aportes del presente trabajo. Por otra parte, en referencia a la categoría “métodos y aplicaciones prácticas” de la taxonomía de Barchini y otros (2007), se señalan los aportes conceptuales incorporados:

- Métodos y aplicaciones prácticas: métodos formales, simulación, métodos cuantitativos, métodos cualitativos, **metodologías para el desarrollo del software**, metodologías para proyectos de software, metodologías ágiles, SCRUM, **modelo del espiral y sistemas de información**.

En Mariño y otros (2010) se expuso una propuesta innovadora basada en vincular la perspectiva dialéctica abordada por Samaja (2004, 2006b) con un proceso del ciclo de vida del desarrollo del software: el modelo en espiral.

En este trabajo, el interés es establecer las primeras relaciones entre la teoría dialéctica, según la posición epistemológica que Samaja (2006a) propone siguiendo a Hegel, con SCRUM una metodología ágil utilizada para la gestión de proyectos software.

1.2. INGENIERÍA DEL SOFTWARE Y METODOLOGÍAS ÁGILES

La ingeniería del software (IS) es una disciplina de la ingeniería que comprende los aspectos de la producción de software desde las etapas iniciales de la especificación del sistema durante todo su ciclo de vida (Pressman, 2005). Este puede ser definido como:

“Un marco de referencia que contiene los procesos, las actividades y las tareas involucradas en el desarrollo, la explotación y el mantenimiento de un producto de software, abarcando la vida del sistema desde la definición de los requisitos hasta la finalización de su uso” (ISO/IEC, 1995).

Existen modelos que lo describen y el orden en que se deben ejecutar las fases componentes. Entre los más tradicionales y utilizados se mencionan: i) Modelo en cascada; ii) Modelo en V; iii) Modelo iterativo; iv) Modelo de desarrollo incremental; v) Modelo en espiral; y vi) Modelo de Prototipos.

En la IS, una metodología permite determinar las tareas a realizar con miras a la mejora del esfuerzo realizado por el equipo de recursos humanos involucrados. Además, la proliferación y empleo de estas, aseguran el logro de los objetivos de un proyecto software, integrando técnicas y métodos en un ciclo completo de realización del proyecto.

Las metodologías tradicionales llevan asociadas un marcado énfasis en el control del proceso mediante una rigurosa definición de roles, actividades y artefactos, incluyendo modelado y documentación detallada, han demostrado ser efectivas y necesarias en proyectos de gran tamaño. Este enfoque no resulta ser el más adecuado para muchos de los proyectos actuales, donde el entorno del sistema es muy cambiante y el tiempo de desarrollo se reduce. En este escenario, las metodologías ágiles emergen como una posible respuesta, por estar especialmente orientadas para proyectos pequeños, constituyen una solución a medida para esos entornos, aportando una elevada



simplificación, pero sin renunciar a las prácticas esenciales para asegurar la calidad del producto (Canós y otros, 2003).

Este estudio preliminar pretende aportar información, con miras a apoyar la idea de que los conceptos de la dialéctica hegeliana y su recuperación desde la perspectiva de Samaja (2006a), se encuentran reflejados en una práctica del desarrollo ágil, en particular en SCRUM.

2. PERSPECTIVA DE SAMAJA RESPECTO A LA DIALÉCTICA HEGELIANA Y SCRUM

En esta sección se sintetizan características relevantes del marco teórico a partir del cual se elabora una propuesta orientada a la vinculación entre aspectos de distintas disciplinas.

2.1. PERSPECTIVA DE SAMAJA RESPECTO A LA DIALÉCTICA HEGELIANA

El proceso descrito como dialéctica hegeliana rechaza el dilema de “deducción-inducción” o “tesis - antítesis” y propone una superación o un paso constructivo al que denomina “síntesis” (Samaja, 2006c). Este paso, en su epigénesis resume la historia anterior, suprimiéndola, conservándola y superándola. Si no se cumplen los objetivos o metas, la síntesis se transforma en el inicio del próximo paso, repitiéndose el proceso: tesis – antítesis – síntesis.

El número de iteraciones que componen este proceso o espiral dialéctica es impredecible. Su evolución se detiene cuando el resultado de una vuelta del espiral o síntesis no genera una nueva tesis y su correspondiente antítesis.

2.2. SCRUM

SCRUM es una metodología para la gestión y control de proyectos, centrada en la construcción de software que satisface las necesidades del cliente, cumple con los objetivos del negocio y el equipo de desarrollo que construye el producto. Al no establecer prácticas de IS, se combina fácilmente con otras metodologías de desarrollo.

Es utilizado por empresas grandes y pequeñas, incluyendo Yahoo, Microsoft, Google, Lockheed Martin, Motorola, SAP, Cisco, GE, CapitalOne y la Reserva Federal de los EE.UU (Deemer y otros, 2010).

Díaz (2009) define a SCRUM como una colección de procesos para la gestión de proyectos, que permite centrarse en la entrega de valor para el cliente y la potenciación del equipo para lograr su máxima eficiencia dentro de un esquema de mejora continua. En Canós y otros (2003) se resumen las principales características de SCRUM:

- Especialmente indicada para proyectos con un rápido cambio de requerimientos.

- El desarrollo de software se realiza mediante iteraciones, denominadas Sprints, con una duración de 30 días. El resultado de cada Sprint es un incremento ejecutable que se muestra al cliente.
- La realización de reuniones a lo largo del proyecto, entre ellas se destaca la diaria del equipo de desarrollo, con una duración aproximada de 15 minutos y con miras a la coordinación e integración de actividades. Los roles, artefactos y eventos principales se resumen en la Figura 1. En la propuesta de Deemer y otros (2009) se indicó el ciclo de realimentación.

Figura 1. Roles, artefactos y eventos principales de SCRUM



Fuente: Deemer y otros (2009).

El proceso SCRUM se compone de iteraciones o Sprint, es donde el Software se lleva a cabo. Al inicio de cada Sprint se selecciona una lista de requerimientos para ser implementados en la próxima iteración y de esta forma progresar hacia el objetivo. Una vez finalizado el Sprint se obtiene del producto, un entregable que muestra y enseña los progresos acometidos en él.

En este punto se procede a una fase de revisión del proyecto para exponer dichos avances e incorporar, si fuera necesario, nuevos requerimientos a la pila de producto (Product Backlog). Estos son seleccionados por el equipo de trabajo, en conjunto con el Scrum Master y el propietario del producto en la reunión de planificación del Sprint.

Tras la realización de cada Sprint se lleva a cabo una reunión retrospectiva que permite aprender de los conocimientos y experiencias adquiridas hasta el momento. Se revisa con el equipo los objetivos marcados inicialmente en el Sprint Backlog, se aplican



los cambios y ajustes si son necesarios, y se marcan los aspectos positivos y negativos, que servirán de realimentación para el paso siguiente.

Paulatinamente, la gestión del proceso con SCRUM refina con los detalles introducidos en la planificación del Sprint e incorpora a través del riesgo cierta incertidumbre que pueda ocurrir durante el desarrollo del mismo. En cada iteración, el riesgo se gestiona en forma continua, a través de las reuniones diarias, de manera informal y ágil con el equipo de desarrollo, de una duración de no más de 15 minutos.

A cada integrante se formulan 3 (tres) preguntas: i) ¿Qué tareas ha realizado desde la última reunión?; ii) ¿Qué tareas va a realizar hoy? y iii) ¿Qué ayuda necesita para realizar esta tarea?, es decir, identificación de obstáculos o riesgos que impidan o pueden impedir la realización de dicha tarea y dificulten el normal avance. Del análisis de riesgo resulta en una decisión de "seguir o no seguir".

A continuación se enuncian algunas prácticas de SCRUM que sustentan a priori la incorporación de los conceptos de la teoría hegeliana, revisada desde la perspectiva de Samaja.

- Establecer los requisitos del producto software plasmados y priorizados en las pilas de productos (Product Backlog). En este caso, en función a los requerimientos del cliente se seleccionan las tareas a incluir en el Sprint.
- Establecer mediante reuniones la viabilidad del proyecto el impacto que puede producir y los problemas que puede causar.
- Gestionar el Sprint: al iniciar cada uno de ellos se evalúa el esfuerzo necesario para llevarlo a cabo, por medio de la estimación de pruebas de Póker o Fibonacci y preparar los contenidos de las nuevas versiones.
- Gestionar el riesgo: en forma continua, a través de las reuniones diarias, las reuniones de revisión y retrospectiva.
- Establecer el alcance del proyecto a través de la pila de productos.
- Agilizar el proceso de desarrollo a través de las entregas frecuentes.
- Comprender mejor el producto software a desarrollar a través de la colaboración y comunicación en el equipo de desarrollo.
- Flexibilizar la modificación del Product Backlog con la incorporación de nuevos requerimientos y la definición de las prioridades al final de cada Sprint.

3. PROPUESTA DE VINCULACIÓN DE LA DIALÉCTICA HEGELIANA DESDE LA PERSPECTIVA DE SAMAJA Y LAS PRÁCTICAS DE SCRUM

Para Langino (1990), citado por Perdomo (2001, p.222), el conocimiento científico "no es el punto final estático de la investigación sino la expresión cognitiva e intelectual de



una continua interacción con nuestro medio natural y social”. Esta construcción del conocimiento desde la teoría epistemológica sustentada, puede verse reflejada en los procesos estudiados por la ingeniería del software.

A continuación se exponen algunas vinculaciones preliminares de SCRUM como método ágil en el cual se refleja la teoría dialéctica desde la perspectiva de Samaja (2006b):

- Es un modo de desarrollo adaptable, antes que predictivo. Al ser adaptable cada Sprint incorpora los cambios recibidos del entorno, generándose el proceso de tesis-antítesis-síntesis.
- Orientado a las personas, más que a los procesos. Los sujetos intervienen en la definición de las actividades a realizar, de las reuniones de trabajo planificadas surgen requerimientos que representan la síntesis de diversas posturas.
- Emplea el modelo de construcción incremental basado en iteraciones y revisiones. Que si se analiza desde una perspectiva epistemológica el mismo refleja el concepto hegeliano abordado por Samaja (2006b).

El software siguiendo a Gilb (1988), citado en Pressman (2005), “evoluciona con el tiempo”. Las metodologías ágiles emplean el modelo de construcción incremental basado en iteraciones y revisiones. Se caracterizan por permitir desarrollar versiones cada vez más completas del mismo. En el caso de SCRUM, esta definición se refleja en el producto generado en cada Sprint. Este conserva y supera al anterior, resguardando en su interior su epigénesis, se está entonces en presencia del proceso dialéctico hegeliano retomado por Samaja.

Es decir, la epigénesis se refleja en cada Sprint de SCRUM. Con cada Sprint se construyen sucesivas versiones del software, cada vez más completas y complejas. De este modo se obtiene un sistema o “todo” operacional. Cada Sprint tiene una historia que es suprimida, conservada y superada hasta cumplimentar los requerimientos del cliente. Esta “síntesis” alcanzada en el desarrollo completo del Sprint, pasa a ser tesis en la próxima iteración.

Es decir, el producto o “tesis” evoluciona nuevamente, ya que como resultado de su interacción con el entorno surgen o se evidencian insatisfacciones del cliente, que pueden ser consideradas como antítesis, provocando nuevas tareas a incluir en la lista de requerimientos, asignándole prioridades a implementarse en el próximo Sprint.

La culminación puede concebirse como la síntesis del modelo dialéctico de Hegel. Toda estructura perseverante -Sprint de SCRUM- es siempre una historia viviente que retorna a su sistema de límites, pero precisamente, por esa propensión al retorno se desarrolla y se transforma (Samaja, 2004). Los requerimientos, se podrían considerar como inductores de una nueva vuelta, es decir, un nuevo proceso dialéctico que en la ingeniería del software se refleja en los sucesivos Sprint generados en el marco del proceso descrito por SCRUM.



4. DISCUSIÓN

La informática como disciplina en incesante evolución y demandada desde diversos sectores de la vida social-cultural-económica, se encuentra en constante revisión y actualización. Así, las relaciones que entretejen sus objetos de estudios (sistemas de información, entre otros) con sus teorías (como la dialéctica) y sus metodologías y procedimientos de tipo técnico (como por ejemplo, SCRUM), se ven permanentemente vinculadas con el cambiante mundo real.

El estudio y análisis de metodologías desempeñan un rol preponderante en el desarrollo de software de calidad. En Mariño y otros (2010) se ilustró la “presencia” de la dialéctica hegeliana en el modelo en espiral, un ciclo de vida de la IS. En este trabajo se ilustró su representación en SCRUM, una metodología para gestionar la producción de software.

Cabe destacar que mientras que en el ciclo de vida del modelo en espiral la “dialéctica hegeliana” se refleja en todo el proceso; en SCRUM este concepto se incorpora en cada Sprint, componente o parte de un modelo para gestionar proyectos.

Un proyecto o la gestión de un proyecto desarrollado aplicando SCRUM puede concebirse como un producto construido a través de un proceso dialéctico. Lo expuesto se fundamenta en la relación sugerida entre la dialéctica hegeliana y el Sprint de SCRUM, que aborda: i) el retorno al origen que fundamenta la idea de síntesis del ciclo y que constituye un nuevo producto o incremento del mismo; ii) la determinación compleja reflejada en los entregables o productos resultantes de cada uno, que surgen a partir de una historia, es decir sus ciclos de los ciclos.

La realimentación lograda al finalizar cada Sprint, se plasma en el proceso dialéctico, donde el equipo está en condiciones de mejorar de forma continua el proceso de desarrollo, visualizándose el proceso de tesis-antítesis-síntesis en cada uno.

CONCLUSIONES

Se propone entonces considerar que la historia de un proyecto de software resulta de la conservación, supresión y mejoras sucesivas, logradas mediante ciclos evolutivos identificados como Sprint, al utilizar SCRUM en la gestión un proyecto. Sus etapas están vinculadas entre sí constituyendo lo que se plantea denominar epigénesis de la gestión de proyectos informáticos.

Es decir, esta gestión podría conceptualizarse como una estructura dinámica, consecuencia de una historia, que es conservada como el fundamento mismo de un proceso de incesante realimentación -los Sprint de SCRUM- se encuentran en constante tensión y movimiento de restitución de la identidad a través de la no identidad. La no identidad indica la necesidad de un nuevo Sprint o ciclo.

A partir de lo expuesto, se propone profundizar en esta línea de investigación y aportar otras vinculaciones entre la ingeniería del software y conceptos hegelianos desde la perspectiva de Samaja.



REFERENCIAS BIBLIOGRÁFICAS

- Aliseda, A. (2007). Emerge una nueva disciplina: las ciencias cognitivas. Documento en línea. Disponible en: <http://www.ejournal.unam.mx/cns/no88/CNS000008803.pdf>. Consultado: 20/01/2011.
- Barchini, G.; Fernández, N. y Lezcano, M. (2007). Modelo curricular de la informática. Revista Iberoamericana de Educación, 42(3).
- Canós, J.; Letelier, P. y Penades, M. (2003). Metodologías Ágiles en el Desarrollo de Software. Documento en línea. Disponible en: <http://www.willydev.net/descargas/prev/ToDoAgil.pdf>. Consulta: 20/01/2011.
- Deemer, P.; Benefield, G.; Larman, C. y Vodde, B. (2010). The Scrum Primer. Version 1.2. Scrum Training Institute. Documento en línea. Disponible en: <http://goodagile.com/scrumprimer/scrumprimer.pdf>. Consulta: 20/01/2011.
- Deemer, P.; Benefield, G.; Larman, C. y Vodde, B. (2009). Información Básica de Scrum the Scrum Primer Version 1.1. Scrum Training Institute. Documento en línea. Disponible en: http://www.goodagile.com/scrumprimer/scrumprimer_es.pdf. Consulta: 30/05/2011.
- Díaz, J. (2009). Las metodologías ágiles como garantía de calidad del software. Revista Española de Innovación, Calidad e Ingeniería del software, Vol.5, Nº 3, 2009.
- International Standard Organization/International Electrotechnical Commission ISO/IEC 12207 (1995). Information Technology - Software life cycle processes. Ginebra.
- Mariño, S.; Godoy, M. y Petris, R. (2010). Conceptos dialécticos reflejados en un proceso de ingeniería del software. Análisis preliminar. Multiciencias, Vol. 10, Nº 3, 2010 (294 - 299).
- Perdomo, M. (2001). Elementos para una filosofía de la ciencia empirista, constructivista, contextual y pragmática. Santiago de Compostela. Edit. Fernando Ramos.
- Pressman, R. (2005). Ingeniería del software. Un enfoque práctico. Mexico D.F. McGraw Hill Interamericana.
- Red de Universidades Nacionales con Carreras en Informática (UNCI) (2006). Propuesta de Currícula RedUNCI. Documento en línea. Disponible en: <http://redunci.info.unlp.edu.ar/docs/Core-basico-23-6-2006-Agosto.pdf>. Consulta: 15/03/2011.
- Samaja, J. (2004). Epistemología de la salud. Reproducción Social, Subjetividad y Transdisciplina. Buenos Aires. Editorial Lugar.
- Samaja, J. (2006a). De una nueva lista de categorías (Introducción y comentarios). II Jornadas del Grupo de Estudios Peirceanos Peirce en Argentina. Documento en línea.



Disponible en: <http://www.unav.es/gep/II/PeirceArgentinaSamaja.pdf> Consulta:
22/11/2010.

Samaja, J. (2006b). Las Ciencias Cognitivas como transdisciplina. Doctorado en Ciencias Cognitivas. Facultad de Humanidades. Buenos Aires. Publicaciones de la Universidad Nacional del Nordeste.

Samaja, J. (2006c). El proceso de la ciencia. Una breve introducción a la investigación científica. Documento en línea. Disponible en:
<http://www.biblioteca.org.ar/libros/131841.pdf>. Consulta: 22/11/2010.