

Revista
Española de
Innovación,
Calidad e
Ingeniería del Software



Volumen 3, No. 2, octubre, 2007

Web de la editorial: www.ati.es

E-mail: reicis@ati.es

ISSN: 1885-4486

Copyright © ATI, 2007

Ninguna parte de esta publicación puede ser reproducida, almacenada, o transmitida por ningún medio (incluyendo medios electrónicos, mecánicos, fotocopias, grabaciones o cualquier otra) para su uso o difusión públicos sin permiso previo escrito de la editorial. Uso privado autorizado sin restricciones.

Publicado por la Asociación de Técnicos en Informática

Revista Española de Innovación, Calidad e Ingeniería del Software (REICIS)

Editores

Dr. D. Luís Fernández Sanz

Departamento de Sistemas Informáticos, Universidad Europea de Madrid

Dr. D. Juan José Cuadrado-Gallego

Departamento de Ciencias de la Computación, Universidad de Alcalá

Miembros del Consejo Editorial

Dr. Dña. Idoia Alarcón

Depto. de Informática
Universidad Autónoma de Madrid

Dr. D. José Antonio Calvo-Manzano

Depto. de Leng y Sist. Inf. e Ing. Software
Universidad Politécnica de Madrid

Dra. Tanja Vos

Instituto Tecnológico de Informática
Universidad Politécnica de Valencia

D. Raynald Korchia

InQA.labs

D. Rafael Fernández Calvo

ATI

Dr. D. Oscar Pastor

Depto. de Sist. Informáticos y Computación
Universidad Politécnica de Valencia

Dra. Dña. María Moreno

Depto. de Informática
Universidad de Salamanca

Dra. D. Javier Aroba

Depto de Ing.El. de Sist. Inf. y Automática
Universidad de Huelva

D. Antonio Rodríguez

Telelogic

Dr. D. Pablo Javier Tuya

Depto. de Informática
Universidad de Oviedo

Dra. Dña. Antonia Mas

Depto. de Informática
Universitat de les Illes Balears

Dr. D. José Ramón Hilera

Depto. de Ciencias de la Computación
Universidad de Alcalá

Contenidos

REICIS

Editorial	4
<i>Luís Fernández Sanz, Juan J. Cuadrado-Gallego</i>	
Presentación	5
<i>Luis Fernández</i>	
Una propuesta organizativa de los procesos de SD y SS en ITIL	6
<i>Magdalena Arcilla; Elena Ruiz, Carlos Cerrada, Gerzón Gómez, José A. Calvo-Manzano, Tomás San Feliu y Ángel Sánchez</i>	
Una revisión sistemática de la adaptación del proceso software	21
<i>Oscar Pedreira, Mario Piattini, Miguel R. Luaces, Nieves R. Brisaboa</i>	
Sección Actualidad Invitada:	39
Servicios Web interoperables y extensibles	
<i>Encarna Quesada, responsable de oficina del W3C en España</i>	

Editorial

The logo for REICIS, consisting of the word "REICIS" in a bold, white, serif font, centered within a solid black rectangular box.

El número de octubre de 2007 de REICIS inaugura una sección de contribuciones invitadas aportadas por expertos de la industria y del mercado que nos transmitirán, en breves reflexiones, las tendencias o los temas de actualidad que perciban como más interesantes desde su posición privilegiada en el ámbito de la ingeniería y la calidad del software. En esta primera aportación es la responsable de la oficina del W3C (World Wide Web Consortium) en España, Encarna Quesada, quien nos informa del estado actual y de las tendencias en las recomendaciones emitidas por esta entidad y que más pueden afectar a los desarrolladores de software. En futuros números, contaremos con otros expertos que nos proporcionarán ideas e información de primera mano sobre las tecnologías e iniciativas emergentes e innovadoras relacionadas con la ingeniería y calidad del software.

Por otra parte, la sección regular de artículos se nutre de dos contribuciones remitidas por sus autores directamente al comité editorial. En este sentido, la revista continua invitando desde estas líneas a todos los profesionales relacionados con el mundo de la Innovación, Calidad e Ingeniería del Software a que utilicen REICIS como el medio para dar a conocer sus trabajos e investigaciones teniendo las máximas garantías de la profesionalidad con que serán tratados sus trabajos. Podrán encontrar todas las instrucciones necesarias para el envío de sus contribuciones en la página web de la revista: www.ati.es/reicis.

Luis Fernández Sanz
Juan J. Cuadrado-Gallego
Editores

Este segundo número de REICIS del año 2007 publica, tras el proceso de revisión de nuestro comité editorial, dos contribuciones remitidas directamente a la revista.

El primer artículo, “Una propuesta organizativa de los procesos de SD y SS en ITIL”, ha sido elaborado en colaboración por un amplio grupo de autores de distintas entidades (Universidad Nacional de Educación a Distancia, Universidad Autónoma de Tamaulipas, Universidad Politécnica de Madrid y Everis). En este trabajo se presenta un estudio de la implantación de las áreas de Service Support y Service Delivery del conocido modelo ITIL (Information Technology Infrastructure Library). A través del análisis de los requisitos de dichas áreas y de las experiencias en organizaciones, se realiza una propuesta de dependencias y relaciones funcionales entre procesos y áreas de actividad.

En la segunda contribución, los autores de la Universidad de la Coruña (Oscar Pedreira, Miguel Luaces y Nieves Brisaboa) junto a Mario Piattini de la Universidad de Castilla La Mancha nos ofrecen “Una revisión sistemática de la adaptación del proceso software”. En este trabajo, se realiza un análisis sistemático de las contribuciones realizadas en las principales publicaciones del área de ingeniería de software sobre la adaptación de los procesos de software a distintas situaciones y organizaciones. El resultado de este análisis permite observar la poca atención prestada a las PYME y la necesidad de un marco general de adaptación de procesos aplicable a un rango mayor de empresas de lo que es habitual actualmente.

Finalmente, en la columna de Actualidad Invitada, es Encarna Quesada, la responsable de la oficina del W3C en España, quien nos ofrece la revisión de los trabajos actuales de esta entidad y sus líneas futuras en el ámbito de los servicios Web y también de XML.

Luis Fernández Sanz

Una propuesta organizativa de los procesos de SD y SS en ITIL

Magdalena Arcilla, Elena Ruiz, Carlos Cerrada

Universidad Nacional de Educación a Distancia, Escuela Técnica Superior de Ingeniería Informática

{marcilla, elena, ccerrada}@issi.uned.es

Gerzón Gómez

Universidad Autónoma de Tamaulipas, Unidad Reynosa-Rodhe

ggomez@uat.edu.mx

Jose A. Calvo-Manzano, Tomás San Feliu

Universidad Politécnica de Madrid, Facultad de Informática

{jacalvo, tsanfe}@fi.upm.es

Angel Sánchez

everis Foundation

Angel.Sanchez@everis.com

Resumen

La gestión de servicios de TI (Tecnologías de la Información) se está convirtiendo en un factor decisivo para el éxito de la mayoría de los negocios. Ya en los años 80 se comenzó en el Reino Unido el desarrollo de un estándar llamado ITIL para dicha gestión. ITIL tiene dos áreas principales de Gestión de Servicios: el Soporte del Servicio, Service Support (SS) y la Provisión del Servicio, Service Delivery (SD). En este artículo se presenta una propuesta organizativa que permite identificar la secuencia de implantación de los procesos de las áreas de SS y SD. Para ello, se utilizan técnicas de teoría de grafos con objeto de representar las relaciones entre los procesos de ITIL, de forma que se encuentren procesos o agrupaciones de procesos fuertemente relacionados. Estas agrupaciones ayudarán a determinar la prioridad de implantación de los procesos de servicios.

Abstract

IT Service Management is becoming a decisive factor for the success of most of the business. During '80s United Kingdom started the development of a standard called Information Technology Infrastructure Library (ITIL). ITIL has two main areas: Service Delivery (SD) and Service Support (SS). In this article an organizational proposal to identify the implementation sequence of the SS and SD processes is presented. In this way, this work uses graphs theory techniques to represent relationships between ITIL processes, in order to find processes or clusters of processes strongly related. These clusters will help to determine the implementation priority of service processes.

Palabras clave: ITIL, Provisión de Servicios, Soporte de Servicios, Gestión de Servicios.

1. Introducción

La Gestión de Servicios de TI (Tecnologías de la Información) se está convirtiendo en un factor decisivo para el éxito o fracaso del negocio en multitud de organizaciones. Una de las causas del aumento de los costes diarios en pérdidas de productividad, de nuevas oportunidades y de clientes, y del aumento de los costes de servicios, se debe a una Gestión de Servicios de TI inadecuada o que no funciona todo lo bien que sería deseable [1].

Aunque la situación no es nueva, y ya en la década de los 80 se comenzó en Inglaterra el desarrollo de un estándar para cubrir esta necesidad, en la actualidad la problemática se está acentuando vertiginosamente. Este estándar auspiciado por el gobierno británico y denominado ITIL (Information Technology Infrastructure Library, Biblioteca de Infraestructura de TI) define un marco genérico de trabajo para la Gestión de Servicios de TI, centrandó su atención de forma especial en la Provisión del Servicio [2] (SD) y en el Soporte del Servicio [3] (SS).

ITIL es uno de los esquemas más utilizados y ampliamente extendidos en lo referente a “mejores prácticas” a la hora de cumplir con requerimientos de gobernabilidad de las TI y con estándares regulatorios, como la ley Sarbanes-Oxley de 2002 [4]. A pesar de que las entidades sometidas al cumplimiento de dicha ley, son aquellas empresas públicas registradas en la Securities and Exchange Comisión (SEC) en los Estados Unidos, la ley Sarbanes-Oxley ha generado mucha expectación a nivel mundial, provocando que multitud de organizaciones del sector de las TI hayan desarrollado normas y controles, basados en ITIL; para cumplir con los criterios de auditoría, control y seguridad de esta ley.

El grado de preocupación actual ha provocado que ITIL haya resurgido como la aproximación más ampliamente aceptada a nivel internacional, y haya sido adoptado por grandes empresas, como IBM [5], Microsoft [6], SUN [7] y HP [8] entre otras, como base para sus soluciones de la Gestión de Servicios. Aunque estas empresas han desarrollado sus propias metodologías de servicios basadas en ITIL, los criterios de implantación seguidos por cada una de ellas, al no estar establecidos explícitamente en ITIL, han sido guiados en cada caso por intereses o experiencias empresariales. A pesar del gran reto planteado, se ha dedicado poco esfuerzo de investigación en la línea de sistematizar los criterios de implantación de los procesos de Gestión de Servicios de TI.

Recientemente están surgiendo nuevas técnicas como los patrones de proceso, o el concepto de componente de proceso que están jugando un papel importante en la sistematización del proceso software y que podrían ser de utilidad en la organización de procesos de servicios. Por ejemplo, el trabajo de Tran Dan Thu [9] realiza un análisis de propiedades topológicas que permite caracterizar componentes de proceso software bien estructurados.

El presente artículo aborda, también desde una perspectiva topológica, la secuencia de implantación de los procesos de Gestión de Servicios definidos en ITIL. Para ello, se utilizan técnicas de teoría de grafos para representar las relaciones existentes entre los procesos de ITIL, con el propósito de encontrar agrupaciones de procesos fuertemente relacionados. Estas agrupaciones ayudarán a determinar la prioridad de implantación de los procesos de servicios.

El artículo está organizado de la siguiente manera. En la sección 2 se realiza una breve descripción sobre cómo se encuentran inicialmente organizados los procesos en ITIL. La sección 3 se dedica a describir el procedimiento desarrollado en este trabajo, con el objeto de poder utilizar este estudio para llegar a determinar la prioridad de implantación de los procesos de ITIL. Por último, la sección 4 recoge las conclusiones obtenidas.

2. Organización y estructura de los procesos de gestión de servicios de TI

Como se ha indicado anteriormente, las dos principales áreas de Gestión de Servicios son SD y SS. ITIL proporciona un conjunto de mejores prácticas para los procesos de Gestión del Servicio de TI, promoviendo un enfoque de calidad para lograr la eficacia y la eficiencia en el uso de los Sistemas de Información.

En la Figura 1, se enumeran los procesos que componen la Gestión de Servicios de TI tanto en el área de SS como en el área de SD.

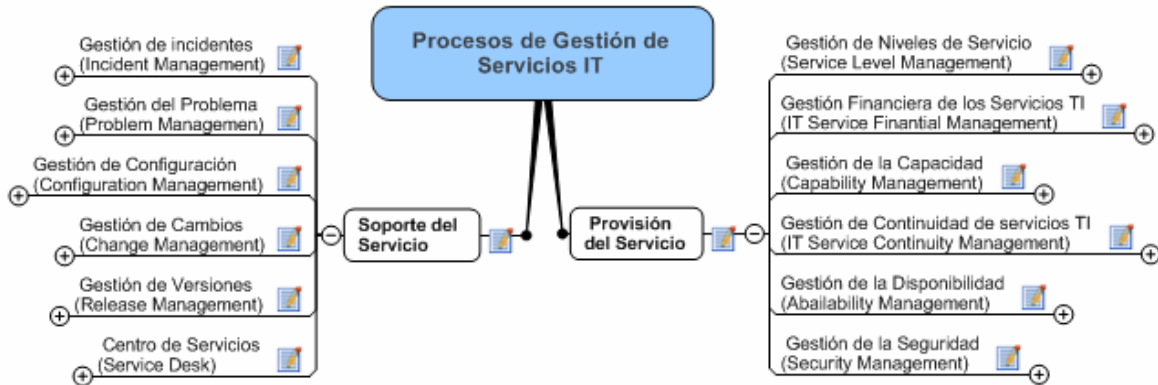


Figura 1. Áreas de la Gestión de Servicios de TI

El área de Soporte del Servicio describe cómo las organizaciones pueden tener acceso a los servicios adecuados para contribuir a su negocio. El área de Provisión del Servicio describe los servicios que necesita el cliente y lo esencial para proporcionar esos servicios. Todos los procesos¹ enumerados anteriormente tienen una estructura común: unos objetivos, el alcance del proceso, los conceptos básicos y la lista de actividades que componen dicho proceso. Algunos procesos definen también los procesos relacionados (aunque en la documentación de ITIL existe un capítulo global [10] [11] que indica por cada proceso cuales son aquellos procesos con los que se relaciona), los costes y los problemas de implantar dicho proceso. La Figura 2 muestra la estructura del proceso de Gestión de Configuración perteneciente al área de Soporte del Servicio.

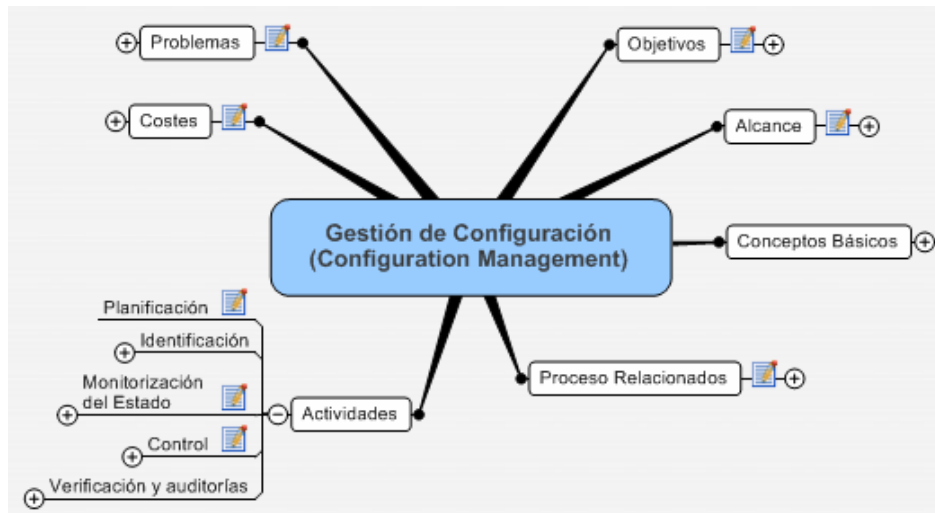


Figura 2. Estructura del proceso de Gestión de Configuración del área de SS

¹La documentación oficial de ITIL considera a Service Desk como una unidad funcional y no como un proceso por lo cual no se analizará en este estudio.

3. Organización de SS Y SD

El procedimiento descrito en esta sección, denominado “PROSSED” (Procedimiento de Organización de Service Support y Service Delivery), permite identificar la secuencia de implantación de los procesos de las áreas de SS y SD, teniendo en cuenta las relaciones entre los procesos [10][11].

La motivación principal para la elaboración del procedimiento surge de la necesidad de disponer de mapas de procesos para determinar la secuencia de implantación de las áreas de proceso SS y SD. PROSSED se divide en dos etapas (véase Figura 3), en la primera se identifican las relaciones entre los procesos dando origen a la matriz de relaciones y, en la segunda, se proponen agrupaciones de Componentes Fuertemente Conexas (CFCs) que dan lugar a la secuencia de implantación.

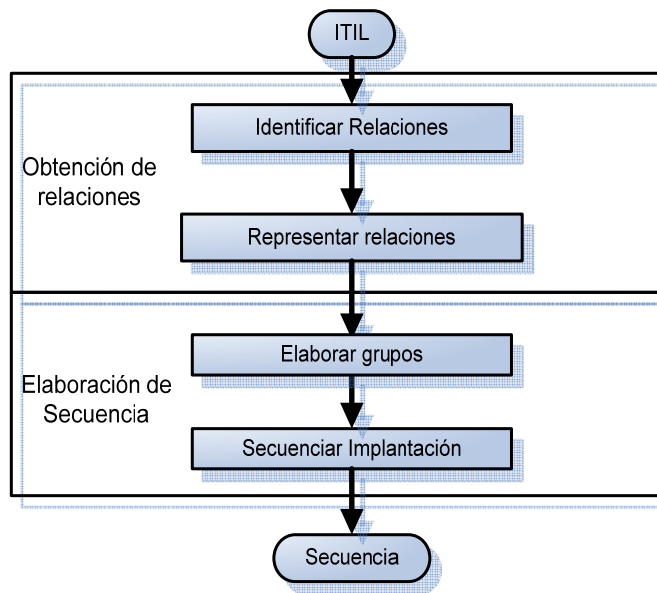


Figura 3. Diagrama de Etapas de PROSSED

3.1 Obtención de relaciones

La primera etapa del procedimiento consiste en la identificación de las relaciones entre los procesos mediante una revisión de la bibliografía oficial de ITIL (capítulo 2 de SS [10] y SD [11]).

3.1.1 Identificación de relaciones

La revisión de la documentación proporciona la información necesaria para determinar con qué procesos se vincula cada proceso. En función de las dependencias encontradas, se elabora una matriz de relaciones (véase Tabla 1), que representa todas las relaciones existentes entre los procesos de las dos áreas de proceso. Para la elaboración de la matriz se han utilizado, en filas y columnas, los acrónimos de los procesos enumerados en la Figura 1 y las expresiones matemáticas $P_i \cap P_j = 1$, si \exists relación entre el Proceso i y el Proceso j , y $P_i \cap P_j = 0$, si no \exists relación entre P_i y P_j , donde TRSS es el número Total de Relaciones entre los procesos de SS, TRSD es el número Total de Relaciones entre los procesos de SD, Total Destino (TD) es el número Total de relaciones entre SS y SD en dirección Destino, y Total Origen (TO) es el número Total de relaciones entre SS y SD en dirección Origen.

	Destino Origen	SS						SD						TD
		IM	PM	CM	ChM	RM	TRSS	SLM	FM	CapM	CoM	AM	TRSD	
SS	IM	1	1	1	1	0	3	1	0	1	0	1		6
	PM	1	1	1	1	0	3	1	0	1	0	1		6
	CM	1	1	1	1	1	4	1	1	1	1	1		9
	ChM	1	1	1	1	1	4	1	0	1	0	1		7
	RM	0	0	1	1	1	2	1	0	0	0	0		3
SD	SLM	1	1	1	1	1		1	1	1	1	1	4	9
	FM	0	0	1	0	0		1	1	1	0	0	2	3
	CapM	1	1	1	1	1		1	1	1	1	1	4	9
	CoM	0	0	1	1	0		1	0	1	1	1	3	5
	AM	1	1	1	1	0		1	0	1	1	1	3	7
	TO	6	6	9	8	4		9	3	8	4	7		

IM: Incident Management	PM : Problem Management	CM: Configuration Management
RM: Release Management	ChM: Change Management	CapM: Capacity Management
SLM: Service Level Management	FM: Financial Management	
CoM: Continuity Management	AM: Availability Management	

Tabla 1. Matriz de relaciones

La Tabla 1 muestra de izquierda a derecha (origen-destino) las áreas de proceso SS y SD con sus respectivos procesos y las dependencias entre ellos. Para cada proceso origen al final de cada fila se ha colocado una celda que contiene el número de relaciones globales

hacia otros procesos (columna TD). Para cada proceso destino al final de cada columna se ha colocado celdas que contienen el número global de relaciones provenientes de otros procesos (fila TO).

3.1.2 Representación de relaciones

Las relaciones mostradas en la Tabla 1 son representadas mediante grafos, primero de forma global (Figura 4) y posteriormente para cada área de proceso (Figura 5 y 6).

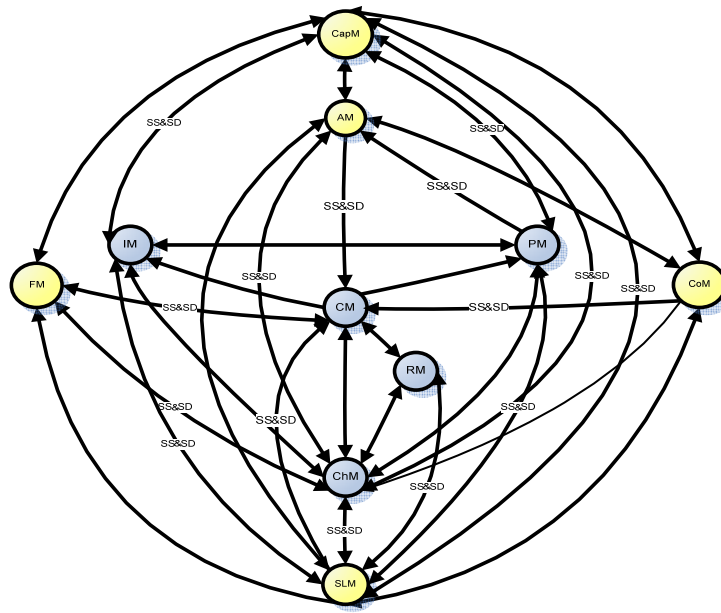


Figura 4. Grafo de los procesos de SS y SD

Los datos proporcionados por la matriz de relaciones se representan en forma de grafo orientado donde los procesos representan los nodos y las flechas representan la existencia de relación. Cada nodo está etiquetado con el acrónimo del nombre del proceso mostrado en la Figura 1.

Las siguientes definiciones describen básicamente la teoría de grafos que se ha empleado:

- Definición 1: Un grafo es un par $G = (V, E) = (V(G), V(E))$, donde V es un conjunto finito no vacío de elementos llamados vértices y E es un conjunto de pares de elementos distintos de V llamados aristas. Es decir, una arista $a \in E$ tiene la forma $\{i, j\}$, donde $i, j \in V$ y $i \neq j$ [12].

- Definición 2: De acuerdo a la definición anterior de grafo, un grafo G es fuertemente conexo si para todo par de vértices i y j ($i \neq j$) existe un camino desde i hasta j y desde j hasta i [13].

Al aplicar las definiciones anteriores a la matriz de relaciones de la Tabla 1 se obtiene la Figura 4 (las etiquetas “SS&SD” indican que la relación corresponde a un proceso de SS con otro de SD o viceversa):

El grafo obtenido en la Figura 4 da una idea de la complejidad de las relaciones entre los procesos debido a que representa todas las relaciones de la Tabla 1. Se observa que no es posible identificar un orden definido para la implantación de los procesos. Por ello, con el objetivo de reducir el nivel de complejidad se agruparán los procesos en áreas de proceso.

3.2 Elaboración de Secuencia

En la segunda etapa de PROSSED se genera la secuencia de implantación de cada área de proceso mediante la elaboración de grupos de procesos de acuerdo a criterios heurísticos.

3.2.1 Elaboración de grupos

Para la elaboración de grupos se utilizan los siguientes criterios heurísticos (CH):

- CH1: Elaborar grupos de procesos por áreas de proceso.
- CH2: Realizar la combinación de procesos C_3^5 de cada área.
- CH3: Obtener agrupaciones cíclicas.

En este paso se particionan los procesos representados en la Figura 4 mediante el primer criterio heurístico (CH1). Posteriormente se elaboran los grafos representados en las Figuras 5 y 6. Los grafos obtenidos se evalúan mediante el software Mathematica y se determina si son o no Componentes Fuertemente Conexas. A las agrupaciones CFCs obtenidas se les aplica el segundo criterio heurístico (CH2). Para obtener las agrupaciones finales se utiliza el tercer criterio heurístico (CH3).

- Partición por áreas de procesos.

Aplicando el primer criterio heurístico (CH1) se obtienen las Figuras 5 y 6, donde se representan los procesos y sus relaciones de cada área de proceso (SS y SD, respectivamente).

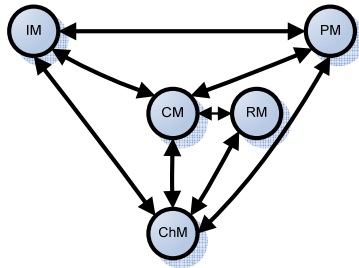


Figura 5. Partición SS

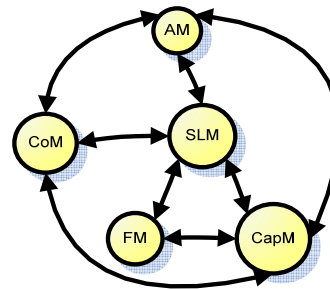


Figura 6. Partición SD

- Verificación de particiones mediante Mathematica.

Las particiones representadas en los grafos de las Figuras 5 y 6 se someten a una evaluación para determinar si son CFCs. Para cada grafo se elabora una componente que recorre todos los vértices (procesos) y arcos (relaciones). Dicha componente es procesada por el Software Mathematica [14] con el objetivo de conocer las CFCs.

- Componente 1: {IM→PM, PM→ChM, ChM→CM, CM→RM, RM→ChM, ChM→IM, IM→CM, CM→PM}
- Componente 2: {AM→CapM, CapM→SLM, SLM→CoM, CoM→AM, AM→SLM, SLM→FM, FM→CapM, CapM→CoM}

La ejecución de las componentes anteriores en Mathematica, obtiene como resultados las siguientes CFCs:

- CFCs de SS: {IM, PM, ChM, CM, RM}
- CFCs de SD: {AM, CapM, SLM, CoM, FM}
- Combinaciones de procesos.

Aplicando el segundo criterio (CH2) se obtienen las Tablas 2 y 3, donde se muestran respectivamente las posibles combinaciones de 3 procesos obtenidas a partir de las particiones SS y SD (siendo 3 el número mínimo posible de componentes para obtener una agrupación cíclica).

Agrupación	Procesos
A	CM, PM, RM
B	CM, PM, ChM
C	CM, PM, IM
D	CM, IM, ChM
E	CM, RM, ChM

Agrupación	Procesos
K	SLM, AM, CoM
L	SLM, FM, CapM
M	SLM, CapM, AM
N	SLM, FM, CoM
O	SLM, CapM, CoM

F	CM, IM, RM
G	IM, PM, ChM
H	IM, PM, RM
I	IM, RM, ChM
J	PM, RM, ChM

Tabla 2. Combinaciones de procesos del área SS

P	SLM, FM, AM
Q	CoM, CapM, AM
R	CoM, FM, AM
S	CoM, FM, CapM
T	AM, FM, CapM

Tabla 3. Combinaciones de procesos del área SD

- Agrupaciones cíclicas.

Una agrupación cíclica es aquella en la que los elementos que la componen son CFCs relacionados directamente.

En las 10 agrupaciones resultantes de cada área, mostradas en las Tablas 2 y 3, se observa que existen agrupaciones que son inviables debido a que están compuestas por procesos cuyas dependencias no son directas y por consecuencia no son cíclicas.

Aplicando el criterio CH3, se obtienen las Tablas 4 y 5.

Agrupación	Procesos
B	CM, PM, ChM
C	IM, PM, CM
D	IM, CM, ChM
E	CM, RM, ChM
G	IM, PM, ChM

Tabla 4. Agrupaciones cíclicas de SS

Agrupación	Procesos
K	SLM, AM, CoM
L	SLM, FM, CapM
M	SLM, CapM, AM
O	SLM, CoM, CapM
Q	CapM, CoM, AM

Tabla 5. Agrupaciones cíclicas de SD

3.2.2 Secuencia de implantación

Las agrupaciones mostradas en las Tablas 4 y 5 indican, en términos de procesos, que ningún proceso podría implantarse sin la puesta en marcha de otro proceso que constituya la agrupación. No obstante para la implantación es necesario identificar el proceso con el cual se iniciará dicha implantación. Para ello, se utilizan los siguientes criterios heurísticos:

- CH4: Número de dependencias dentro del área de proceso (Columnas TRSS y TRSD en Tabla 1).

Las Tablas 6 y 8 muestran la misma información que las columnas TRSS y TRSD de la Tabla 1. Las Tablas 7 y 9 muestran el orden de los procesos en base al número de dependencias dentro de cada área (CH4); por ejemplo se puede observar la agrupación E de la Tabla 4 donde CM=4, ChM=4 y RM=2 según la Tabla 6. Si el orden de implantación se realiza según el número de dependencias entonces debe iniciarse con el proceso CM (agrupación E1) o ChM (E2). La existencia de estas dos posibilidades se resuelve con la implementación del criterio heurístico CH5.

Agrupación	TRSS
IM	3
PM	3
CM	4
ChM	4
RM	2

Tabla 6. Dependencias entre los procesos SS

Agrupación	TRSD
SLM	4
CapM	4
CoM	3
AM	3
FM	2

Tabla 8. Dependencias entre los procesos SD

Agrupación	Orden Implantación		
	1°	2°	3°
B1	CM	ChM	PM
B2	ChM	CM	PM
C1	CM	IM	PM
C2	CM	PM	IM
D1	CM	ChM	IM
D2	ChM	CM	IM
E1	CM	ChM	RM
E2	ChM	CM	RM
G1	ChM	IM	PM
G2	ChM	PM	IM

Tabla 7. Posibilidades de orden de implantación de cada agrupación SS

Agrupación	Orden Implantación		
	1°	2°	3°
K1	SLM	CapM	FM
K2	CapM	SLM	FM
L1	SLM	CoM	AM
L2	SLM	AM	CoM
M1	CapM	CoM	AM
M2	CapM	AM	CoM
O1	SLM	CapM	AM
O2	CapM	SLM	AM
Q1	SLM	CapM	CoM
Q2	CapM	SLM	CoM

Tabla 9. Posibilidades de orden de Implantación de cada agrupación SD

- CH5: Número de dependencias globales de la Figura 4 (columna total-destino (TD) y fila total-origen (TO) en Tabla 1).

El uso del criterio CH5 requiere conocer el número global de dependencias que cada proceso tiene con los demás en las áreas de proceso SS y SD. Dicha información se resume en las Tablas 10 y 11, las cuales se obtienen a partir de la matriz de relaciones (véase Tabla 1, columna total-destino (TD) y fila total-origen (TO)).

	IM	PM	CM	ChM	RM
Origen	6	6	9	7	3
Destino	6	6	9	8	4
Sumatoria	12	12	18	15	7

Tabla 10: Número de dependencias globales de los procesos del área SS

	SLM	FM	CapM	CoM	AM
Origen	9	3	9	5	7
Destino	9	3	8	4	7
Sumatoria	18	6	17	9	14

Tabla 11: Número de dependencias globales de los procesos del área SD

La Tabla 10 indica el número de dependencias que salen (fila origen) y entran (fila destino) de los procesos SS hacia otros procesos (que pueden ser de SS o SD), y la Tabla 11 indica el número de dependencias que salen (fila origen) y entran (fila destino) de los procesos de SD hacia otros (que pueden ser de SS o SD). Al aplicar el criterio CH5, se observa que, por ejemplo en el área SS, el proceso con mayor número de dependencias es CM y el siguiente es ChM (véase fila Sumatoria).

La Tabla 12 muestra el orden de implantación de las agrupaciones cíclicas de SS, resultado de la aplicación del criterio CH4 y CH5. Dado que cada empresa posee necesidades particulares, cada una seleccionará el orden que requiera, no obstante esta investigación hace la recomendación mostrada en la Tabla 14.

Agrupación	Orden Implantación		
	1°	2°	3°
E1	CM	ChM	RM
C1	CM,	IM	PM
G1	ChM	IM	PM
B1	CM	ChM	PM
D1	CM	ChM	IM

Tabla 12. Secuencia de implantación de las agrupaciones de SS

De forma similar a la expuesta anteriormente se obtienen las secuencias de implantación para los procesos del área de SD (Tablas 13 y 15).

Agrupación	Orden Implantación		
	1°	2°	3°
L1	SLM	CoM	AM
K1	SLM	CapM	FM
M1	CapM	CoM	AM
O1	SLM	CapM	AM
Q1	SLM	CapM	CoM

Tabla 13. Secuencia de implantación de las agrupaciones de SD

- CH6: Número de dependencias globales de la Figura 4 (sumatoria de columna total-destino y fila total-origen para cada proceso).

La Tabla 14 es el resultado de aplicar nuevamente el criterio CH6 sobre la Tabla 12. Se observa en la Tabla 10, para la sumatoria de dependencias de procesos, que el que más dependencias posee es CM, le sigue ChM, después IM o PM y finalmente RM. Para la situación donde IM = PM, situación única en las 2 áreas de proceso, se ordena en base a su funcionalidad. De acuerdo a ITIL –SS [3], para que exista un problema primero debe existir un incidente. Por lo tanto, debe implantarse primeramente el proceso IM y después PM.

Secuencia de Implantación de SS					
orden	1°	2°	3°	4°	5°
proceso	CM	ChM	IM	PM	RM

Tabla 14. Secuencia de implantación de SS

Secuencia de Implantación de SD					
orden	1°	2°	3°	4°	5°
proceso	SLM	CaPM	AM	CoM	FM

Tabla 15. Secuencia de Implementación de SD

4. Conclusiones

Se observa que todos los procesos de las áreas de proceso Soporte del Servicio (SS) y Provisión del Servicio (SD) están fuertemente conectados, lo que confirma la organización

funcional por áreas de proceso. No obstante, en cuanto a la organización estructural se ha encontrado que no todos los procesos poseen la misma estructura, lo que dificulta identificar las actividades de cada proceso, así como la forma de interrelacionarse entre si. Esta situación abre la posibilidad de estudios futuros que propongan una organización estandarizada de la estructura de los procesos.

ITIL propone un conjunto de procesos que han de implantarse pero no su secuencia de implantación. Este trabajo ha planteado desde el principio, mediante dependencias entre procesos, criterios heurísticos y justificaciones formales, el proceso que permite iniciar la secuencia de implantación en las áreas de proceso Soporte del Servicio y Provisión del Servicio. Para el área Soporte del Servicio se ha encontrado que debe iniciarse por Gestión de la Configuración (Configuration Management) y para el área de Provisión del Servicio con Gestión de Niveles de Servicio (Service Level Management).

De acuerdo a la consultora **Everis** suele presentarse tres escenarios típicos de implantación de ITIL en el ámbito empresarial: 1) Empezar por Gestión de Configuración, 2) Empezar por Centro de Servicios, 3) Empezar por Gestión de Niveles de Servicio. La secuencia de implantación obtenida mediante PROSSED ha sido contrastada con los resultados de la consultora Everis demostrando que efectivamente los resultados obtenidos se corresponden con una secuencia actual de implantación de los procesos ITIL.

Debido a que cada empresa tiene sus propias necesidades y prioridades PROSSED se podría aplicar para obtener la secuencia de implantación de clúster de 3 procesos en lugar de los clúster de 5 procesos obtenidos. En este estudio, por áreas de proceso, se han encontrado 10 agrupaciones de 3 procesos cada una donde 5 agrupaciones corresponden a Soporte del Servicio y 5 corresponden a Provisión del Servicio. Cada agrupación está fuertemente relacionada de tal forma que no debería implementarse un proceso sin la puesta en marcha de los demás.

Los resultados obtenidos representan dependencias a nivel de proceso por área de proceso. Para estudios futuros se considerarán las relaciones entre áreas de proceso así como el grado de vinculación que tienen unos procesos con otros con objeto de poder determinar agrupaciones más específicas y altamente relacionadas que permitan elaborar una estrategia de implantación óptima.

Agradecimientos

Este trabajo está parcialmente patrocinado por las empresas ENDESA, Everis Foundation y SUN Microsystems a través de la “Cátedra de Mejora del Proceso Software en el Espacio Iberoamericano”. Así como por la Secretaría de Educación Pública (México) con una beca PROMEP a través del convenio con la UAT.

Referencias

- [1] Johnson, B.; <http://www.ca.com/hk/event/itil2005/bjohnson.htm>, Noviembre 2006.
- [2] Office of Government Commerce (OGC), *ITIL Managing IT Service: Service Delivery*, TSO, London, 2001.
- [3] Office of Government Commerce (OGC), *ITIL Managing IT Service: Service Support*, TSO, London, 2001.
- [4] Sarbanes, P.; Oxley, M.; *Ley Sarbanes – Oxley (SOX / SORBOX)*, Estados Unidos, Julio 2002.
- [5] IBM, <http://www-306.ibm.com/software/tivoli/features/ITIL/>, Noviembre 2006.
- [6] Microsoft, <http://www.microsoft.com/technet/itsolutions/cits/mo/mof/default.msp>, Noviembre 2006.
- [7] SUN, <http://es.sun.com/services/itil/>, Noviembre 2006.
- [8] HP, <http://www.hp.com/large/itsm/>, Noviembre 2006.
- [9] Thu, T.D.; Hanh, N.; Bich Thuy, D.T.; Coulette, B., Cregut, X.; Topological Properties for Characterizing Well-formedness of Process Components; *Software Process Improvement and Practice*, Volume 10, Issue 2, pp 217-247, 2005.
- [10] Office of Government Commerce (OGC), *Chapter 2: Relationship between processes en “ITIL Managing IT Service: Service Delivery”*, TSO, London, 2001.
- [11] Office of Government Commerce (OGC), *Chapter 2: Relationship between processes en “ITIL Managing IT Service: Service Support”*, TSO, London, 2001.
- [12] Sarabia Viejo, A.; *La investigación operativa*, UPCO, Madrid, 1996.
- [13] Diestel, R.; *Graph Theory*, Springer-Verlag, New York, 1997.
- [14] WolframResearch, Inc., *Matemática 5.2*, <http://www.wolfram.com/mathematica/functions/advanceDocumentationGraphPlot>, Noviembre 2006.

Una revisión sistemática de la adaptación del proceso software

Oscar Pedreira¹, Mario Piattini², Miguel R. Luaces¹, Nieves R. Brisaboa¹

¹Laboratorio de Bases de Datos, Facultad de Informática, Universidade da Coruña
{opedreira, luaces, brisaboa}@udc.es

²Grupo Alarcos, Escuela Superior de Informática, Universidad de Castilla-La Mancha
Mario.Piattini@uclm.es

Abstract

Although software process proposals appear continuously, it is difficult to fit any of them into a given company as they are. Thus, some kind of adaptation or tailoring is always necessary. The goal of software process tailoring is to adapt an “off-the-shelf” software process to meet the needs of a specific organization or project. Although process tailoring is a mandatory activity in most software process proposals, it is usually carried out by following an ad-hoc approach, and the amount of research done on this topic to date can be considered small. This paper presents a systematic review of software process tailoring, analyzing the existing approaches towards this activity, discussing the main issues related to the problem, and providing an up-to-date and complete framework in which to position new research activities.

Resumen

Aunque continuamente aparecen nuevos trabajos y propuestas en el área de proceso software, es difícil que encajen en su forma original en una empresa dada. De ahí la necesidad de adaptar los procesos estándar a las características particulares de la empresa. El objetivo de la adaptación del proceso software (*software process tailoring*) es adecuar un proceso software estándar a las características de una organización o proyecto específico. Aunque la adaptación del proceso software es señalada como una actividad obligatoria por la mayoría de las metodologías, en general se lleva a cabo siguiendo a cabo un enfoque *ad-hoc*, y la cantidad de investigación realizada en este tema puede considerarse pequeña. Este artículo presenta una revisión sistemática de la adaptación del proceso software, analizando las aproximaciones actuales para esta actividad, discutiendo las cuestiones más importantes relacionadas con este problema, y proporcionando un marco de trabajo actualizado en el que posicionar nuevas investigaciones.

Palabras clave: Adaptación del proceso software, revisión sistemática

1. Introducción

El proceso software es una de las áreas de investigación más importantes para la comunidad de ingeniería del software. Continuamente aparecen nuevos trabajos y propuestas que definen distintas aproximaciones para el proceso de desarrollo de software. Sin embargo, es difícil que satisfagan todas las necesidades de una organización o proyecto específico. Teniendo en cuenta que dos organizaciones son diferentes entre si y que, incluso dentro de una misma organización, dos proyectos pueden ser también muy diferentes, el proceso aplicado con éxito en uno de ellos puede ser un completo fracaso en el otro. Por eso, el proceso software debe ser adaptado al contexto y características específicas de cada caso.

La adaptación del proceso software (en inglés, *software process tailoring*) consiste en adaptar y particularizar la descripción general del proceso para obtener un nuevo proceso adaptado, aplicable en un entorno alternativo y probablemente menos general [1]. Es decir, adaptar un proceso software a las necesidades concretas de una organización o un proyecto dado. La adaptación del proceso software puede tener lugar en dos niveles diferentes: a nivel organizacional o a nivel de proyecto.

Las consecuencias de una mala adaptación del proceso software pueden ser muy importantes para la organización. En primer lugar, factores como el presupuesto, tiempo de desarrollo y calidad del producto dependen directamente de la adecuación del proceso software a los proyectos. Un proceso software mal adaptado a la empresa/proyecto puede incluir actividades innecesarias que suponen una pérdida de tiempo y dinero, o la omisión de algunas necesarias, que puede afectar a la calidad del producto. Además, una mala adaptación del proceso software puede dar lugar a problemas con respecto a la conformidad con estándares como ISO 9000 [2] o CMMI [3]. Por último, y no por eso menos importante, la adaptación del proceso software también influye en la satisfacción del personal, pues perder tiempo en actividades innecesarias no suele ser muy motivador.

Hoy en día, la adaptación del proceso software es considerada como una actividad necesaria por la mayoría de las organizaciones de desarrollo, pero suele llevarse a cabo sin la suficiente dedicación, siguiendo un enfoque *ad-hoc* y sin ningún tipo de regla o guía. Aunque se han propuesto aproximaciones sistemáticas y formales para la adaptación del proceso software, la investigación llevada a cabo en esta área hasta la fecha puede considerarse pequeña. En la literatura existente podemos encontrar estudios que proponen

herramientas de soporte para la adaptación del proceso software. También trabajos que presentan experiencias reales en la adaptación de metodologías conocidas a situaciones específicas, muy interesantes debido a la naturaleza práctica de este problema.

En este trabajo se ha llevado a cabo una revisión sistemática de la investigación en adaptación del proceso software. Para esto, se obtuvieron y analizaron los trabajos de investigación más importantes en el tema para identificar las aproximaciones, métodos y herramientas de soporte para la adaptación del proceso software. El resto del artículo está organizado como sigue. La siguiente sección describe brevemente el concepto de revisión sistemática en ingeniería del software. La sección 3 describe las decisiones tomadas en cada paso de la revisión sistemática presentada en este artículo, y los resultados a que han dado lugar. La sección 4 presenta las cuestiones más importantes relacionadas con la adaptación del proceso software identificadas en la literatura. Finalmente, la sección 5 presenta las conclusiones y trabajo futuro.

2. Revisión sistemática en ingeniería del software

La revisión sistemática es un método de investigación desarrollado para obtener, analizar, y evaluar toda la investigación relevante para una pregunta de investigación o un área de interés particular [4]. En contraste con una revisión literaria tradicional, una revisión sistemática sigue una secuencia estricta y bien definida de pasos metodológicos, que garantizan el alto valor científico de los resultados obtenidos. La principal razón para llevar a cabo una revisión sistemática es incrementar la probabilidad de detectar más resultados reales en el área de interés que los obtenidos con una revisión menos formal. Una revisión sistemática requiere un esfuerzo considerablemente mayor en comparación con una revisión tradicional, pero este es el precio a pagar por una revisión profunda y completa de un área de interés determinada.

El concepto de revisión sistemática apareció en el área de la medicina, y su adaptación a la ingeniería del software se presenta en [5]. El método propuesto consta de tres actividades principales: planificación, revisión y publicación. Durante la actividad de planificación se identifican las necesidades de la revisión y se desarrolla el protocolo de revisión. En la actividad de revisión se seleccionan y evalúan los estudios primarios más importantes para ese área de investigación. El último paso consiste en la publicación de los

resultados obtenidos en la revisión. Para que la revisión sistemática sea más sencilla, en [4] se propone una plantilla para el protocolo de revisión, que es la que se ha seguido en este trabajo.

3. Una revisión sistemática de la adaptación del proceso software

Esta sección presenta el desarrollo de cada fase de la revisión sistemática (formulación de la pregunta, selección de las fuentes, selección de estudios primarios y extracción de información), y los resultados obtenidos en cada una de ellas.

3.1. Formulación de la pregunta

El objetivo de esta revisión sistemática es obtener y analizar todas las herramientas, técnicas, aproximaciones y experiencias propuestas para la adaptación del proceso software para a situación determinada. También analizar los trabajos que tratan cuestiones relacionadas con la adaptación del proceso software y la conformidad con estándares como ISO 9000 [2] o CMMI [3]. Por último, también consideramos interesante analizar los trabajos que presentan experiencias reales adaptando procesos software como el proceso unificado de desarrollo o extreme programming, en empresas de desarrollo de software. Anteriores revisiones sistemáticas en otras áreas de investigación se concentran en problemas más particulares pero, debido a la pequeña cantidad de investigación disponible en éste área, se decidió analizar toda la investigación disponible en adaptación del proceso software.

Al contrario que la plantilla de protocolo propuesta en [4], esta revisión sistemática debe contestar varias preguntas y no sólo una debido al amplio alcance que trata de cubrir:

- ¿Cuáles son las principales aproximaciones, métodos, y herramientas para la adaptación del proceso software?
- ¿Cuáles son las guías más conocidas para la adaptación del proceso software teniendo en cuenta el problema de la conformidad con estándares?
- ¿Hay disponibles casos de estudio sobre experiencias reales en la adaptación del proceso software?

El objetivo perseguido con estas preguntas era proporcionar un informe lo más amplio y comprensible posible del estado del arte en la adaptación del proceso software al finalizar la

revisión sistemática. No sólo queríamos identificar las principales aproximaciones en esta área, si no también sus puntos fuertes y debilidades y, por supuesto, el trabajo futuro que puede llevarse a cabo para solventar estas debilidades.

De estas preguntas se extraen directamente las palabras clave utilizadas para construir las consultas emitidas contra las fuentes de estudios primarios. Estas palabras son (en inglés): *software, process, tailoring, methodology, unified process, extreme programming, tool, improvement, organization, practice, experience, case study, standard, compliance, ISO, CMM*. Durante la búsqueda de estudios primarios se consideró importante comprobar si sinónimos de la palabra *tailoring*, como *adaptation* o *customization* se usan también, como puede ocurrir con *method* como sinónimo de *process*, pues esto puede ser una fuente de problemas en la búsqueda de estudios primarios.

3.2. Selección de fuentes

Las consultas utilizadas en la búsqueda de estudios primarios se construyeron a partir de la lista de palabras clave anterior. La cadena de búsqueda básica fue “*software process tailoring*”. También se utilizaron cadenas de búsqueda adicionales que incluyen otras palabras clave. La consulta “*(software AND (process OR method) AND (tailoring OR adaptation OR customization))*” se utilizó para comprobar si los sinónimos identificados anteriormente realmente afectan a los resultados de la búsqueda. La cadena “*software process tailoring AND (practice OR experience OR organization)*” se utilizó con la intención de obtener posibles trabajos sobre casos de estudio en empresas de desarrollo de software reales.

La búsqueda de estudios primarios se llevó a cabo consultando las bibliotecas digitales de las organizaciones y editoriales más relevantes para la comunidad de ingeniería del software. La lista de fuentes iniciales es la siguiente: IEEE Computer Science Digital Library, ACM Digital Library, Wiley Interscience (área de computación), Science@Direct (área de computación), y SpringerLink. Por supuesto, antes de comenzar la revisión, se partió de algunos estudios importantes acerca de adaptación del proceso software. Quizá los más importantes sean [9] y [1]. Ambos presentan un marco de trabajo bien estructurado para la adaptación del proceso software sin suponer la utilización de ninguna metodología en particular, y se consideraron un buen punto de partida por lo que fueron incluidos directamente en la revisión.

3.3. Selección de estudios primarios

Para la obtención de los estudios primarios se consultaron las fuentes citadas en la sección anterior con las cadenas de consulta. Con la consulta “*software process tailoring*” se obtuvieron la mayoría de los resultados seleccionados. Con las demás consultas se lograron tan sólo dos resultados que no pudieron ser recuperados anteriormente debido a los problemas con los sinónimos “*tailoring*” y “*adaptation*”. Las demás cadenas de búsqueda no devolvieron ningún estudio primario adicional. Estos datos junto con el pequeño número de resultados obtenidos demuestran la poca investigación llevada a cabo en el tema hasta la fecha.

La mayoría de los resultados devueltos por las fuentes de estudios primarios no eran de interés para el objetivo de esta investigación. El criterio de inclusión/exclusión utilizado para seleccionar los estudios primarios relevantes se aplicó sobre el título y resumen de cada estudio. En algunos casos esto no fue suficiente y el texto completo del estudio tuvo que ser parcialmente revisado para asegurarse de que realmente era relevante para el objetivo de la revisión. Durante la selección de trabajos, se tuvo en cuenta que algunos resultados aparecen en más de una fuente, por lo que los estudios repetidos se identificaron y controlaron. Por supuesto, también se tuvieron en cuenta los artículos citados en los estudios primarios, y todos aquellos relativos a la adaptación del proceso software también fueron incluidos en el conjunto de estudios primarios.

Tras consultar las fuentes de datos, se recuperaron un total de 394 estudios no repetidos. Una vez aplicado el criterio de inclusión/exclusión a cada uno de ellos, sólo 28 se consideraron relevantes para las preguntas que esta revisión trata de contestar. La Tabla 1 muestra el número total de resultados y estudios primarios obtenidos de cada fuente. La lista de resultados primarios seleccionados se incluye por completo al final de este artículo. El número total de estudios primarios es pequeño en comparación con los resultados obtenidos en revisiones sistemáticas realizadas en otras áreas de interés. Sin embargo, el reducido número de estudios primarios no es un resultado malo para la revisión sistemática, puesto que su objetivo es encontrar y analizar toda la investigación existente en esta área de interés. Por otra parte, se puede considerar una evidencia que la investigación disponible hasta la fecha en este tema es pequeña y que, en principio, todavía queda trabajo por hacer.

Fuente	Resultados	Estudios primarios
ACM	200	3
IEEE	100	14
Springer	31	6
Science@Direct	10	2
Wiley	50	0
Otros	3	3
Total	394	28

Tabla 1. Número total de resultados y estudios primarios obtenidos de cada fuente.

Quizá otra observación interesante es que dos de los estudios que superaron los criterios de inclusión/exclusión durante la fase de extracción de información no trataban realmente el problema de la adaptación del proceso software. Sin embargo, sólo dos errores en un total de 394 artículos puede considerarse un buen resultado.

3.4. Extracción de la información.

Una vez identificados y obtenidos los estudios primarios, la siguiente fase consistió en su revisión y en la extracción de la información que estos ofrecen. Para cada uno de ellos, se redactó un resumen de la revisión como el propuesto en [4], describiendo la metodología seguida por los autores del estudio primario, posibles problemas encontrados en el estudio, y una evaluación subjetiva del mismo. Esta evaluación subjetiva consistió básicamente en un pequeño resumen en el que se comparaban las ideas propuestas en el estudio con las propuestas en otros similares.

4. Cuestiones relativas a la adaptación del proceso software.

Como ya hemos mencionado, el objetivo de la adaptación del proceso software es partir de un proceso software estándar y adaptarlo a las necesidades de una organización o proyecto determinados. Hay muchos factores que influyen en esta adaptación, como el tamaño de la organización, sus objetivos, los recursos disponibles, tipo de proyecto, entorno del cliente, tipo de negocio, tecnologías utilizadas, requisitos del cliente sobre el propio proceso de desarrollo, etc. Esta sección presenta las respuestas a las preguntas formuladas en la revisión sistemática. Durante la evaluación y análisis de los estudios primarios se

identificaron como relevantes las siguientes cuestiones relacionadas con la adaptación del proceso software:

- La adaptación del proceso software puede tener lugar a dos niveles distintos en una empresa: adaptación a nivel de organización o a nivel de proyecto.
- La adaptación del proceso software puede llevarse a cabo siguiendo un enfoque formal o informal.
- Casos de estudio sobre experiencias en organizaciones reales.
- Adaptación del proceso software en PYMEs frente a grandes empresas.
- Consideración de los problemas relacionados con la conformidad con estándares.
- Descripción de herramientas de soporte para la actividad de adaptación del proceso.

Cada estudio primario puede tratar una o más de estas cuestiones. Por ejemplo, algunos artículos describen una aproximación formal para la adaptación del proceso software y una experiencia real aplicando su propuesta en una empresa, pero trabajando sólo a nivel de proyecto.

La Figura 1 muestra el porcentaje de artículos que se centran en una aproximación formal o informal, al nivel de proyecto u organización, y en pequeñas o grandes empresas. La figura da una idea del esfuerzo dedicado a cada problema. Por ejemplo, la mayoría de los artículos revisados describen un caso de estudio en grandes empresas de desarrollo de software más que en pequeñas y medianas empresas.

La tabla 2 resume los estudios primarios que tratan cada uno de estos aspectos en la adaptación del proceso software, aunque a lo largo de esta sección se explicarán con más detalle.

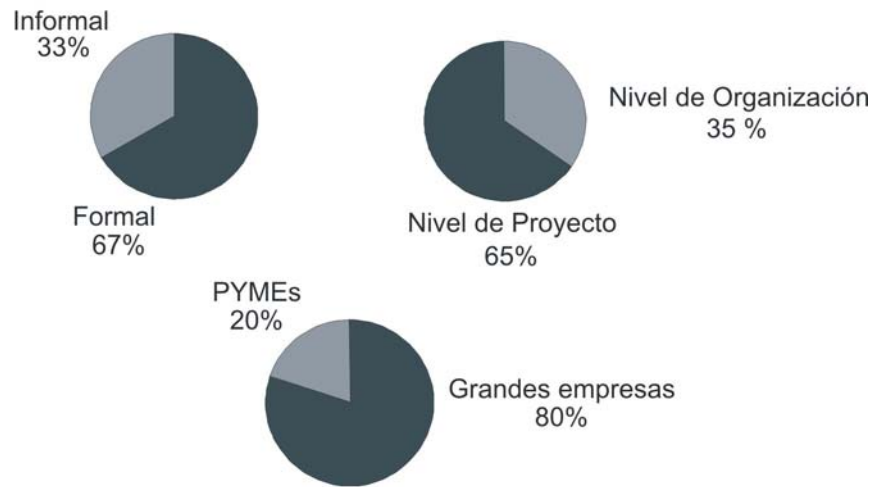


Figura 1. Porcentaje de estudios dedicados a cada tópic.

Nivel organizacional	[1], [12], [15], [16], [17], [24], [26], [29]
Nivel de proyecto	[1], [6], [7], [8], [9], [10], [11], [13], [14], [19], [20], [21], [27], [30], [32]
Aproximación formal	[1], [6], [7], [8], [9], [11], [19], [24], [26], [28], [29], [30]
Aproximación informal	[10], [12], [14], [15], [16], [17]
PYME	[10], [12], [25], [27]
Gran empresa	[1], [6], [7], [8], [9], [11], [13], [14], [15], [16], [17], [19], [29]
Caso de estudio	[6], [8], [10], [12], [13], [14], [15]
Herramienta	[20], [21], [22]
Conformidad	[1], [19], [31]

Tabla 2. Resumen de referencias en cada cuestión relativa a la adaptación del proceso software.

4.1. Distintos niveles para la adaptación del proceso software.

La adaptación del proceso software puede tener lugar en diferentes niveles. Por ejemplo, estudios como [6] [7] [8] distinguen entre adaptación del proceso software a nivel de la organización y a nivel de proyecto. La adaptación de proceso a nivel organizacional consiste en adaptar un proceso software estándar para definir el proceso software de la organización (por ejemplo, definir el proceso software en un SGC basado en ISO 9000 adaptando RUP). La adaptación a nivel de proyecto consiste en la adaptación de ese proceso software de la organización a las necesidades particulares que pudieran darse en un

proyecto determinado. El objetivo de la adaptación del proceso software a nivel de organización es adaptar un proceso software estándar a las necesidades concretas de la organización². Este proceso está adaptado a las necesidades y el contexto específico de esa empresa. Algunos elementos del proceso de partida se eliminan en la versión adaptada porque no son necesarios, pero en ocasiones hay que añadir otros, debido al tipo de sistemas que desarrolla la empresa.

Sin embargo, esto puede no ser suficiente y la adaptación del proceso software puede ser necesaria también a nivel de proyecto. Es decir, el proceso estándar de la organización (OSSP) debe ser adaptado a las necesidades específicas de cada proyecto. Teniendo en cuenta que dentro de una misma empresa un proyecto puede ser muy diferente de otro, el proceso aplicado con éxito en uno de ellos puede dar lugar a malos resultados en el otro. Como señala [6], los errores cometidos al alinear el proceso software estándar de la empresa con el contexto del proyecto puede tener consecuencias en el tiempo de desarrollo, la calidad y el coste del proyecto. Este problema es mayor para empresas en las que los proyectos son realmente grandes, costosos y muy diferentes entre si. En casos como éste, la adaptación del proceso software a las características del proyecto es algo casi obligatorio. Para este tipo de organizaciones, [9] es un buen ejemplo de guías de adaptación para satisfacer las necesidades de cada proyecto. Seguir el enfoque de adaptar el proceso a cada proyecto en particular también permite la utilización de una aproximación totalmente diferente para el desarrollo de software en cada proyecto si es necesario.

Como podemos ver en la figura 1, la mayoría de los estudios revisados consideran la adaptación del proceso software sólo al nivel de proyecto. Sin embargo, la adaptación a nivel de organización también es necesaria, y las guías y reglas para esta adaptación pueden ser bastante diferentes en este caso (pues deben considerar a la empresa en su conjunto, y no el contexto limitado de un proyecto). Algunos trabajos van incluso más lejos. Si en la empresa se distinguen varios tipos bien diferenciados de proyectos, se puede considerar otro nivel más en la adaptación del proceso software. En este caso, el OSSP se adapta para cada uno de esos tipos de proyectos. Sin embargo, sólo [10] distingue este nivel de adaptación, que puede ser demasiado para empresas que no son realmente grandes.

² El proceso resultante suele denominarse (en inglés) “Organizational Software Standard Process” (OSSP) [6].

Como conclusión, podemos decir que el nivel de detalle de la adaptación del proceso tanto a nivel organizacional como a nivel de proyecto, depende del tamaño de la empresa y sus proyectos. La adaptación del proceso software a nivel de organización debería considerarse siempre como una necesidad. Para algunas empresas con líneas de productos claramente diferenciadas, podría considerarse la adaptación del proceso software a las necesidades particulares de cada una de ellas. Finalmente, como se menciona en [10], la adaptación del proceso podría tener lugar sólo a nivel de proyecto si este es muy grande y la adaptación representa sólo un pequeño porcentaje del tiempo total del proyecto.

4.2. Aproximaciones formales e informales para la adaptación del proceso software

La distinción entre una aproximación formal/sistemática o una adaptación informal también es importante. Algunos de los trabajos revisados proponen seguir algún tipo de aproximación formal, en forma de proceso definido que comprende todos los pasos necesarios para la adaptación sistemática del proceso software. Sin embargo, muchos otros trabajos presentan una adaptación informal del proceso para una situación determinada, basándose en guías o recomendaciones. El grado de formalidad en los diferentes estudios revisados puede ser muy diferente entre unos y otros. Por ejemplo, la aproximación propuesta en [6] es bastante informal si la comparamos con las propuestas en [9] o [11].

Los estudios como [6] presentan un método ordenado, sistemático y muy completo para la adaptación del proceso software. La principal ventaja de una aproximación sistemática es que el resultado no depende tanto de la capacidad personal del responsable de esta actividad ni de sus preferencias personales en cuanto a metodologías de desarrollo de software. Un enfoque de este tipo proporciona un marco de trabajo bien estructurado para analizar todos los factores relevantes para el proceso de desarrollo. Esto puede ser de gran ayuda cuando el responsable de esta actividad no tiene mucha experiencia en este tema o si la organización es compleja. Este enfoque puede ser una buena opción para organizaciones de desarrollo de software grandes que puedan asumir el esfuerzo de este grado de formalidad. Sin embargo, para una empresa pequeña un enfoque con un nivel de formalidad tan alto puede ser excesivo. Por ejemplo, [10] presenta una experiencia en una pequeña empresa y concluye que en una situación como esta la adaptación será mejor

siguiendo un proceso sencillo y pragmático y no como un proceso estrictamente planificado y gestionado.

4.3. Casos de estudio: experiencias en organizaciones reales

Algunos de los estudios revisados presentan su propuesta junto con los resultados obtenidos al aplicarla en una organización real de desarrollo de software (como [6][12], por ejemplo). Otros presentan un caso real siguiendo una aproximación informal para la adaptación del proceso. La mayoría de los estudios que siguen una aproximación informal describen su experiencia en la adaptación del proceso unificado o Programación Extrema a situaciones específicas. Por ejemplo, [13] y [14] describen sus experiencias adaptando XP a proyectos grandes/complejos en grandes organizaciones de desarrollo de software, mientras que [15] presenta una experiencia en el uso de una versión adaptada del proceso unificado de desarrollo.

Como señala [8], la ausencia de investigación basada en la práctica en el desarrollo de software en general, y en la adaptación del proceso software en particular, es sorprendente en un campo aplicado. Los casos de estudio han demostrado ser una herramienta potente en otras áreas y el estudio de experiencias reales también puede ser una importante fuente de información para el desarrollo de nuevas aproximaciones en esta área.

Otro problema que encontramos en los casos de estudio sobre la adaptación del proceso software es que la mayoría de los estudios presta más atención al proceso resultante que a las guías, métodos y criterios seguidos para la adaptación del proceso.

4.4. Adaptación del proceso software en pequeñas o grandes empresas

Quizá este es uno de los resultados más interesantes derivados de la revisión. La mayoría de los estudios revisados proponen un método de adaptación definido para grandes organizaciones de desarrollo de software. Por ejemplo, [7][8][13] describen el método de adaptación del proceso software que se siguió en una gran empresa de telecomunicaciones. [11] presenta una metodología cuantitativa para la adaptación del proceso software en un entorno de alto riesgo como el NASA/SEL y [16][17] hacen lo mismo para los laboratorios de Raytheon.

La adaptación del proceso software es muy importante para este tipo de organizaciones, pero todas ellas están certificadas al menos hasta el nivel 3 de CMM y son

capaces de asimilar esta adaptación compleja mejor que una pequeña empresa. Una aproximación tan fuerte puede ser demasiado pesada para una PYME. La adaptación del proceso software es normalmente una actividad difícil en este tipo de empresas ya que no suelen tener un entorno de desarrollo de software tan bien organizado como las grandes organizaciones y porque la experiencia y los recursos de los responsables de esta actividad suelen ser más limitados.

Pocos estudios se centran en el caso de las pequeñas empresas; [10] describe cómo, en su experiencia, la adaptación del proceso software en PYMES se hace mejor con un proceso sencillo y pragmático y no con uno demasiado formal y estricto, [12] es un estudio realmente interesante centrado en pequeñas empresas, que propone una aproximación ligera. Describe la experiencia aplicándolo en dos pequeñas empresas incluyendo resultados cuantitativos sobre el proceso resultante. Quizá esta es una de las carencias más importantes en la investigación en la adaptación del proceso software.

4.5. Adaptación del proceso software y conformidad con estándares

La adaptación del proceso software presenta un problema interesante en lo que respecta a la conformidad con estándares. En organizaciones con una certificación ISO 9000 [2] o CMMi [3], la adaptación del proceso software debe asegurar que el proceso resultante es conforme a dichos estándares. A nivel organizacional, la adaptación para crear el proceso software de la empresa debe asegurar que el proceso es conforme a los requisitos de los estándares. A nivel de proyecto, la adaptación del proceso organizacional debe asegurar también que el resultado es conforme a dichos requisitos. Se trata de una cuestión importante ya que, si no se tiene en cuenta, la adaptación del proceso podría dar lugar al no cumplimiento de esos requisitos.

La adaptación del proceso software es una actividad obligatoria en modelos como CMM [18] y CMMI [3]. En [1] proporciona un marco de trabajo completo y guías para la adaptación del proceso software en empresas que siguen estos modelos. Éste es uno de los estudios más completos sobre adaptación del proceso software y uno de los pocos que trata el problema de la conformidad con estándares. Otro de los artículos que trata este problema es [19] que propone un marco de trabajo estrictamente formal para la adaptación del proceso software e introduce la idea de verificar automáticamente el proceso resultante

midiendo el número de dependencias entre elementos del mismo que preserva. [31] también aborda este problema con un enfoque formal.

4.6. Herramientas de soporte para la adaptación del proceso software

Como es habitual, el uso de herramientas de apoyo puede facilitar la actividad de adaptación del proceso software. Estudios como [20] y [21] definen la adaptación del proceso software como una actividad intensiva en conocimiento y analizan los beneficios del uso de herramientas de gestión del conocimiento en este tipo de tareas. Distinguen entre el uso de conocimiento general sobre la adaptación del proceso software y el uso de conocimiento contextualizado sobre experiencias previas en la empresa. Como demuestran estos estudios, el uso de este tipo de conocimiento puede ser de gran ayuda, principalmente para personal con poca experiencia que debe enfrentarse a esta tarea.

[22] presenta una herramienta diferente que se basa en el uso de redes neuronales para la adaptación semiautomática del proceso software utilizando registros históricos como datos de aprendizaje para su sistema. Por tanto esta herramienta también se basa en el uso de conocimiento contextualizado en las actividades de adaptación del proceso software, aunque la ausencia de experiencias previas puede ser un problema para obtener buenos resultados de la misma.

5. Conclusiones

Este artículo presenta una revisión sistemática de la adaptación del proceso software que recoge y analiza la investigación más importante llevada a cabo en este tema hasta la fecha. El artículo proporciona un marco de trabajo actualizado en el que posicionar nuevas actividades de investigación. La aproximación sistemática seguida para la realización de esta revisión garantiza la compleción de los resultados obtenidos.

Como resultado de la revisión se identificaron distintas cuestiones de interés en la problemática de la adaptación del proceso software. Uno de los problemas más importantes que discutimos es el grado de formalidad en la actividad de adaptación del proceso software. Algunos de los estudios primarios revisados proponen un marco formal que demostró funcionar con éxito en situaciones reales. Sin embargo, todas estas experiencias tuvieron lugar en organizaciones de desarrollo de software grandes y avanzadas que pueden hacer frente a esta formalidad. En empresas pequeñas y medianas, una aproximación de

este tipo puede ser excesiva. Así, normalmente este tipo de empresas se enfrentan a la adaptación del proceso software siguiendo un enfoque ad-hoc, lo que hace que el proceso resultante sea muy dependiente de la capacidad y preferencias de la persona responsable. Así, el proceso software obtenido finalmente puede no ser el más adecuado a las características de la empresa.

Como conclusión, la carencia más importante que hemos identificado en la adaptación del proceso software es la poca atención prestada a las PYMEs, y la necesidad de dedicar más esfuerzo al desarrollo de un marco general para la adaptación del proceso software aplicable a un espectro mayor de empresas.

En nuestra opinión, la conformidad con estándares es otra área en la que todavía queda trabajo por hacer. Este problema es vital para empresas certificadas ISO [2] o CMMI [3]. Como la desviación del proceso estándar de la organización puede suponer problemas importantes para conservar la certificación, en ocasiones la falta de un marco para la adaptación del proceso software puede forzar a estas empresas a seguir un proceso que comprende actividades innecesarias para un proyecto determinado. La investigación disponible en esta área es realmente limitada y el problema está todavía abierto.

Uno de los problemas que encontramos es que las aproximaciones existentes para la adaptación del proceso software fueron definidas pensando en entornos específicos. Así, nuestro trabajo futuro en esta línea de investigación incluye el desarrollo de un marco completo y general para la adaptación del proceso software, aplicable a un mayor rango de empresas.

Agradecimientos

Este trabajo fue parcialmente financiado por: Xunta de Galicia PGIDIT05SIN10502PR and 2006/4, Ministerio de Educación y Ciencia (PGE y FEDER) TIN2006-15071-C03-03 (para Oscar Pedreira, Miguel Luaces y Nieves Brisaboa); el Programa FPU del Ministerio de Educación y Ciencia (AP-2006-03214) (para Óscar Pedreira); el proyecto ESFINGE del Ministerio de Educación y Ciencia (FEDER) (TIN2006-15175-C05-05) (para Mario Piattini) y el proyecto COMPETISOFT (CYTED- 506PI0287).

Referencias

[1] Ginsberg, M., Quinn, L., *Process tailoring and the software Capability Maturity Model*. Technical report, Software Engineering Institute (SEI), USA, 1995

- [2] ISO 9001:2000. *Quality management systems. Requirements*. International Organization for Standardization, 2000
- [3] *CMMI for Systems Engineering/Software engineering. Version 1.1*. Technical report, Software Engineering Institute (SEI), 2002
- [4] Biolchini, J., Mian, P.G., Natali, A.C.C., Travassos, G.H., *Systematic review in software engineering*. Technical report, Systems Engineering and Computer Science Department, UFRJ, Brasil, 2005
- [5] Kitchenham, B., *Procedures for performing systematic reviews*. Technical report Software Engineering Group, Department of Computer Science, Keele University, 2004
- [6] Nanda, V., “On tailoring an organizational standard software development process for specific projects”. En: *Proceedings of the 11th International Conference on Software Quality*, pp. 1-13, 2001
- [7] Fitzgerald, B., Russo, N., OKane, T., “An empirical study of system development method tailoring in practice”. En: *Proceedings of the Eighth European Conference on Information Systems*, pp. 187-194, 2000
- [8] Fitzgerald, B., Russo, N., O’Kane, T., “Software development method tailoring at Motorola”. *Communications of the ACM*, vol. 46, num. 4, pp. 65-70, 2003
- [9] Budlong, F., Szulewski, P., Ganska, R., *Process tailoring for software project plans*. Technical report, Software Technology Support Center of the U.S. Air Force, USA, 1996
- [10] Hanssen, G.K., Westerheim, H., Bjrnson, F.O., “Tailoring RUP to a defined project type: A case study”. En: *Product Focused Software Process Improvement: 6th International Conference, PROFES 2005*, pp. 314-327. Lecture Notes in Computer Science (3547/2005), Springer, 2005.
- [11] Basili, V., Rombach, D., “Tailoring the software process to project goals and environments”. En: *Proceedings of the 9th international conference on Software Engineering*, pp. 345-357, IEEE CS Press, 1987
- [12] von Wangenheim, C.G., Weber, S., Hauck, J.C.R., “Experiences on establishing software processes in small companies”. *Information and Software Technology* vol. 48, pp. 890-900, Elsevier, 2000
- [13] Bowers, J., May, J., Melander, E., Baarman, M., Ayoob, A., “Tailoring XP for large system mission critical software development”. En: *XP/Agile Universe 2002: Second XP*

Universe and First Agile Universe Conference, pp. 100-111. Lecture Notes in Computer Science (2418/2002), Springer, 2002

[14] Cao, L., Mohan, K., Xu, P., Ramesh, B., “How extreme does extreme programming have to be? Adapting XP practices to large-scale projects”. En: *Proceedings of the 37th Hawaii International Conference on System Sciences*, pp. 1-10, IEEE CS Press, 2004

[15] Westerheim, H., Hanssen, G.K., “The introduction and use of a tailored unified process a case study”. En: *Proceedings of the 2005 31st EUROMICRO Conference on Software Engineering and Advanced Applications (EUROMICRO-SEAA05)*, pp. 196-203, IEEE CS Press, 2005

[16] Oshana, R., “An industrial application of cleanroom software engineering – benefits through tailoring”. En: *Thirty-First Annual Hawaii International Conference on System Sciences*, vol. 6, pp. 122-131. IEEE Press, 1998

[17] Oshana, R.S., “Tailoring cleanroom for industrial use”. *IEEE Software*, pp. 46-55, IEEE Press, 1998.

[18] *CMM: Capability Maturity Model*. Technical report, Software Engineering Institute (SEI), 1993

[19] Yong, I.C., Min, S.Y., Bae, D.H., “Tailoring and verifying software process”. En: *Proc. Eighth Asia-Pacific Software Engineering Conference (APSEC'01)*, pp. 202- 209, IEEE Press, 2001

[20] Xu, P., Ramesh, B., “A tool for the capture and use of process knowledge in process tailoring”. En: *Proceedings of the 36th Hawaii International Conference on System Sciences (HICSS03)*, IEEE Press, 2003

[21] Xu, P., “Knowledge support in software process tailoring”. En: *Proceedings of the 38th Hawaii International Conference on System Sciences*, pp.1-9, IEEE CS Press, 2005

[22] Park, S., Naa, H., Parka, S., Sugumarana, V., “A semi-automated filtering technique for software process tailoring using neural network”. *Expert Systems with Applications*, vol. 30, pp. 179-189, Elsevier, 2006

[23] Kitchenham, B.A., Peeger, S.L., Hoaglin, D.C., Emam, K.E., Rosenberg, J., “Preliminary guidelines for empirical research in software engineering”. *IEEE Transactions on Software Engineering*, vol. 28, num. 2, pp. 721-734, IEEE Press, 2002

- [24] Hikichi, K., Fushida, K., Iida, H., Ichi Matsumoto, K., “A software process tailoring system focusing to quantitative management plans”. En: *Product-Focused Software Process Improvement, 7th International Conference, PROFES 2006*, pp. 441-446, Lecture Notes in Computer Science (4034/2006), Springer, 2006
- [25] Brodman, J.G., Johnson, D.L., “A software process improvement approach tailored for small organizations and small projects”. En: *Proceedings of the International Conference on Software Engineering (ICSE 97)*, pp. 661-662, ACM Press, 1997
- [26] Keenan, F., “Agile process tailoring and problem analysis (APPLY)”. En: *Proceedings of the 26th International Conference on Software Engineering (ICSE04)*, pp. 45-47, ACM Press, 2004
- [27] Kim, S.Y., Choi, H.J., “An evaluation of process performance for a small-team project-a case study”. En: *Proceedings of the Fourth Annual ACIS International Conference on Computer and Information Science (ICIS05)*, pp. 308-313, IEEE CS Press, 2005
- [28] Bustard, D.W., Keenan, F., “Strategies for systems analysis: Groundwork for process tailoring”. In: *Proceedings of the 12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS05)*, pp. 357- 362, IEEE Press, 2005
- [29]. Hollenbach, C., Frakes, W., “Software process reuse in an industrial setting”. En: *Fourth International Conference on Software Reuse (ICSR'96)*, pp. 22- 30, IEEE Press, 1996
- [30] Lobsitz, R.M., “A method for assembling a project-specific software process definition”. In: *29th Hawaii International Conference on System Sciences (HICSS'96) Volume 1: Software Technology and Architecture*, pp. 722-730, IEEE Press, 1996
- [31]. Welzel, D., Hausen, H.L., Schmidt, W., “Tailoring ASRID conformance testing of software processes: The ProcePT approach”. In: *2nd IEEE Software Engineering Standards Symposium*, IEEE Press, 1995
- [32] Henninger, S., Baumgarten, K., “A case-based approach to tailoring software processes”. En: *Proceedings of the 4th International Conference on Case-Based Reasoning, ICCBR 2001*, pp. 249-262, Lecture Notes in Computer Science (2080/2001), Springer, 2001

- [33] Mnkandla, E., Dwolatzky, B., Mlotshwa, S., “Tailoring agile methodologies to the southern african environment”. In: *Extreme Programming and Agile Processes in Software Engineering*, pp. 259-262, Lecture Notes in Computer Science (3556/2005), Springer, 2005
- [34] Pikkarainen, M., Salo, O., “A practical approach for deploying agile methods”. In: *Extreme Programming and Agile Processes in Software Engineering.*, pp. 213-214, Lecture Note in Computer Science (4044/2006), Springer, 2006

Servicios Web interoperables y extensibles

Encarna Quesada

Responsable de la Oficina del W3C en España
Edificio CTIC
Parque Científico Tecnológico de Gijón
Cabueñes s/n
33203-Gijón
equesada@w3.org

Introducción

El W3C (World Wide Web Consortium) [1], es un consorcio internacional formado por empresas y organizaciones de todo el mundo, que trabajan conjuntamente con el objetivo de “Guiar la Web hacia su máximo potencial a través del desarrollo de protocolos y pautas que aseguren el crecimiento futuro de la Web”.

El W3C ha llevado a cabo ese objetivo mediante la creación de estándares y directrices aplicadas a la Web desde 1994, desarrollando durante este tiempo más de 100 estándares, también conocidos como Recomendaciones W3C [2].

La labor del W3C en el área de Servicios Web, principalmente a través de la Actividad de Servicios Web del W3C [3], es destacable por el desarrollo alcanzado en la comunicación entre aplicaciones en la web. Esta actividad centra su trabajo en el diseño de la infraestructura, la definición de la arquitectura y el desarrollo de tecnologías que permitan crear servicios web interoperables y extensibles.

El Lenguaje de Etiquetado Extensible (XML) [4] juega un papel primordial en el proceso de intercambio, estructuración y envío de datos en la Web. Se trata de un formato de texto simple y muy flexible, que permite, entre algunas de sus características, la publicación electrónica internacionalizada e independiente del medio; ahorrar costes gracias a la utilización de herramientas de propósito general para el procesamiento de datos que no presentan costes excesivos; ahorrar costes en desarrollo y formación, ya que

proporciona un formato único que sirve para una gran variedad de usos; aumentar la fiabilidad, debido a que los agentes de usuario pueden automatizar en mayor medida el procesamiento de los documentos que reciben; proporcionar las bases de la Web Semántica, permitiendo un nuevo nivel completo de interoperabilidad e intercambio de información; favorecer la definición por parte de las empresas de protocolos independientes de plataforma para el intercambio de datos, incluyendo el comercio electrónico; facilitar que la gente pueda controlar la manera en la que la información se muestra gracias a las hojas de estilo; y hace posible la reutilización de datos a largo plazo sin dependencias de herramientas propietarias o formatos indocumentados.

XML es muy utilizado como modelo para la definición de nuevos formatos de intercambio de información, por lo que el número de aplicaciones de XML crece rápidamente, y se prevé que seguirá creciendo.

La actividad de Servicios Web del W3C

La Actividad de Servicios Web del W3C ha puesto en marcha diferentes grupos de trabajo e interés como son el Grupo de Trabajo de Anotaciones Semánticas para la Descripción de Servicios Web, Grupo de Trabajo de Direccionamiento de Servicios Web, Grupo de Trabajo de Coreografía de Servicios Web, Grupo de Trabajo de Directivas de Servicios Web, Grupo de Trabajo de Protocolo de XML, Grupo de Trabajo de Modelos Esquema XML para la Vinculación de Datos y Grupo de Interés de servicios web semánticos. Estos grupos han generado ya 7 Recomendaciones y dos borradores de trabajo en el área de servicios web.

El objetivo principal de esta actividad es generar mecanismos que valgan como base para la creación de aplicaciones basadas en Servicios Web. Estas tecnologías permiten, desde la creación y envío de los mensajes, hasta la posibilidad de realizar definiciones de sus funcionalidades o de componer varios servicios para que se comporten como una unidad; definiciones requeridas para la búsqueda o descubrimiento de los servicios demandados.

Un ejemplo de protocolo base y que tuvo además gran éxito ya que se convirtió en una tecnología fundamental para los servicios web fue SOAP [5], infraestructura de mensajería basada en XML. SOAP 1.2 se convirtió en Recomendación en Junio de 2003. En Enero de

2005, MTOM (Mecanismos de optimización de transmisión de mensajes SOAP) pasó a Recomendación.

El futuro de los Servicios Web

La evolución del mercado hace que surjan nuevas tecnologías basadas en servicios web con el objetivo de resolver temas concretos. Las nuevas especificaciones en las que trabaja el W3C hacen posible la creación de entornos cada vez más evolucionados. Por ejemplo, WSDL (Web Services Description Language) [6] es un lenguaje de descripción de servicios Web que permite definir la funcionalidad abstracta y la forma de acceder a un servicio. Ya que los protocolos de comunicación y los formatos de los mensajes se están estandarizando en la comunidad web, se va haciendo cada vez más importante la descripción de las comunicaciones de forma estructurada. WSDL satisface esa necesidad al definir una gramática XML para describir servicios de red como colecciones de agentes de comunicación capaces de intercambiar mensajes.

WSDL se ha extendido a través de la labor del Grupo de Trabajo de Anotaciones Semánticas para WSDL, constituido en abril de 2006, con el objetivo de elaborar un soporte para añadir semántica a los servicios web, permitiendo de esta forma obtener descripciones semánticas.

WSDL 2.0 no ofrece la posibilidad de incluir semántica en las definiciones de los servicios web, de esta forma puede ocurrir que dos servicios tengan descripciones sintácticamente similares pero con un significado completamente distinto. A través de la extensión desarrollada por el Grupo de Trabajo de Anotaciones Semánticas para WSDL se está elaborando un soporte genérico para añadir semántica a los Servicios Web. La posibilidad de incluir descripciones semánticas puede ser de gran utilidad en el momento en el que por ejemplo, un consumidor de servicios desea hacer una búsqueda de un servicio que realiza determinada funcionalidad, ya que podría asignársele el servicio que más se adecua a sus requerimientos de forma precisa en el caso de que hubiese más de uno con descripciones sintácticas similares. Esto podría hacerse mediante razonamientos que busquen similitudes entre la semántica de la solicitud de un servicio y su descripción.

Por otro lado, la especificación Anotaciones Semánticas para WSDL y Esquema XML (SAWSDL) [7], que pasó a propuesta de recomendación el 5 de julio de 2007, desarrollada

por este grupo, define mecanismos utilizando anotaciones semánticas que pueden ser añadidas a los componentes WSDL. SAWSDL no especifica un lenguaje para describir modelos semánticos como por ejemplo ontologías. En su lugar, proporciona mecanismos a través de los cuales conceptos de los modelos semánticos definidos tanto dentro como fuera del documento WSDL pueden ser referenciados dentro de los componentes WSDL como anotaciones. Estas semánticas cuando son expresadas en lenguaje formal pueden resolver la ambigüedad en la descripción de los servicios web durante la búsqueda automática y la composición de los servicios web.

Referencias

- [1] World Wide Web Consortium (W3C) <<http://www.w3.org/>>
- [2] Informes Técnicos del W3C <<http://www.w3.org/TR>>
- [3] Actividad de Servicios Web del W3C <<http://www.w3.org/2002/ws/Activity>>
- [4] Lenguaje de Etiquetado Extensible <<http://www.w3.org/XML/>>
- [5] SOAP <<http://www.w3.org/2000/xp/Group/>>
- [6] Lenguaje de Descripción de Servicios Web <<http://www.w3.org/2002/ws/desc/>>
- [7] Anotaciones Semánticas para WSDL y Esquema XML
<http://www.w3.org/2002/ws/sawSDL/>

Perfil profesional



Encarna Quesada Ruiz es responsable de la Oficina Española de W3C. Licenciada en Sociología por la Universidad Complutense de Madrid, especializada en nuevas tecnologías, anteriormente trabajó en diferentes empresas en el área de internacionalización, globalización y localización.