



13th Conference on Software Engineering and Databases

XIII Jornadas de Ingeniería del Software y Bases de Datos

Gijón (Spain), October 7-10 2008

EDITORS: Ana Moreira
María José Suárez-Cabal
Claudio de la Riva
Javier Tuya

**13th Conference
on Software Engineering
and Databases**

**XIII Jornadas
de Ingeniería del Software
y Bases de Datos**

Gijón (Spain), October 7-10 2008

EDITORS: Ana Moreira
María José Suárez-Cabal
Claudio de la Riva
Javier Tuya

Edita:
Ana Moreira
María José Suárez-Cabal
Claudio de la Riva
Javier Tuya

Filmación e impresión:
Gráficas Rigel

Depósito Legal:
AS - 5.236 - 08

ISBN:
978-84-612-5820-8

Volume Editors Details

Ana Moreira

Departamento de Informática
Faculdade de Ciências e tecnologia
Universidade Nova de Lisboa
2829-516 Caparica, Portugal
E-mail: amm@di.fct.unl.pt
URL: <http://ctp.di.fct.unl.pt/~amm/>

María José Suárez-Cabal

Departamento de Informática
Universidad de Oviedo
33204 Gijón, Spain
E-mail: cabal@uniovi.es

Claudio de la Riva

Departamento de Informática
Universidad de Oviedo
33204 Gijón, Spain
E-mail: claudio@uniovi.es
URL: <http://www.di.uniovi.es/~claudio/>

Javier Tuya

Departamento de Informática
Universidad de Oviedo
33204 Gijón, Spain
E-mail: tuya@uniovi.es
URL: <http://www.di.uniovi.es/~tuya/>

Preface

Celebrating 13 Years of JISBD

With the 2008 edition in Gijón (October 7 to 10), the Conference on Software Engineering and Databases (JISBD) celebrates 13 years of existence. Born as a forum where the Spanish community would publish their work, meet to discuss potential research collaborations and evaluate the progress of research projects funded by the Spanish Ministry of Science and Technology, JISBD has long since moved beyond its initial boundaries and crossed several oceans.

Presently, the conference has become an important reference for younger researchers, as well as a forum which the more experienced do not wish to miss. In recent years, JISBD has broadened its radius, accepting papers also in English and Portuguese, in addition to Spanish. This change, not only brought more conference participants, but also significantly increased the number of submissions and, principally, the quality of the submissions accepted.

The JISBD community is now self-sustained and continues to expand. The quality of work accepted is equivalent to that of other relevant international events. In recent years, it has been possible to edit a special volume of IEEE LA with extended versions of the best conference papers and this also is happening with the current edition. This special issue, together with the conference proceedings with ISBN, is a showcase of the quality of the work of JISBD.

One of the highlights of this conference has been the excellence of its keynote speakers. Many of the most admired international researchers and professionals have already been invited to address the JISBD participants.

Within this rich framework for scientific and technological interchange, the conference includes several satellite events. In addition to the presentation of high quality original papers in the main conference, the program includes tutorials, tool demonstrations and workshops for the discussion of innovative ideas and work in progress, as well as a forum to bring to a wider audience research work already published in prestigious journals or conference proceedings (with an acceptance rate below 25% and an impact factor above 0.5).

It is no exaggeration to claim that JISBD has been consolidating its position as a reference event where researchers and professionals of Software Engineering and Databases can get together to discuss results and share ideas. JISBD has become an important forum for collaboration between different strands and research groups, while continuing to offer its participants a well organized event with exceptional hospitality.

About this Edition

The increased global reach of JISBD is evident in the origin of papers received. This year, in addition to the two Iberian and ten Latin-American countries, submissions arrived also from China, France, Germany, India, Iran and Pakistan.

Of a total of 115 abstracts, 112 papers were submitted for review. Most papers were reviewed by three PC members, and several were reviewed by four. The program Committee accepted 30 full papers and selected 12 for presentation as short papers. The acceptance rate for full papers was approximately 25%.

The increasing success of the conference implies greater responsibilities in terms of guaranteeing independent judgement and ensuring compliance with international standards of ethics. For this reason, a greater effort has been made in recent years to avoid double submissions, a task made

more difficult by the fact that the conference accepts submissions in three languages. This year, three good papers were rejected due to double submission, in different languages to different events.

In addition to the accepted papers, the conference includes five workshops, one tutorial, nine tool demos, an industrial panel and also a forum to discuss important relevant work already published elsewhere.

A highlight of the conference is, without doubt, the excellence of the invited keynote speakers. This year is no exception and we are honoured indeed to receive Bashar Nuseibeh and Bran Selic.

Bashar Nuseibeh is an academic and researcher at the Open University in the UK and invited professor in various other universities, including Japan's National Institute of Informatics. Bashar chairs several international committees and is recognized also for industry work, including organizations such as the UK's National Air Traffic Services (NATS), Texas Instruments, Praxis Critical Systems, Philips Research Labs, and NASA.

Bran Selic was, for many years, a distinguished engineer and researcher at IBM, and currently heads a global consultancy based in Canada. He is internationally known for his work in large-scale industrial systems, and for his pioneering work in Model-Driven Development and Real-Time Embedded Systems.

Bashar's keynote is entitled "*The five W's (and one "H") of Security: ... Software Engineering of Secure Systems*" while Bran's is on "*Model-Based Software Engineering: Expected and Unexpected Challenges*".

Acknowledgements

A very special word of thanks is due to Bashar and Bran for having accepted my invitation and for sharing all the participants their knowledge, experience and refined wit. I sincerely hope JISBD was also for them a gratifying and unique experience.

Acknowledgements are due to a multitude of collaborators without whom the conference could not have been a success. Firstly, the paper authors for the trust placed in the quality of JISBD as a conference that merited their submissions. Secondly, to the PC members, whose diligent review work ensured that the authors' trust continues to be justified.

For managing the submission and review process, I was fortunate to have the constant help of Juan Hernández and José Javier Berrocal; they were my guardian angels, constantly alert to deadlines and ready to help as necessary. A special thanks for the contribution of my "Executive Program Committee", Antonio Vallecillo, Juan Hernández, Miguel Toro, Vicente Pelechano and Xavier Franch.

Acknowledgement is due to the main conference organizers, especially to Javier Tuya and his co-chair, Claudio de la Riva, for their efficient handling of the numerous tasks that a conference of this size and quality entails. Thanks also to those responsible for the satellite events (in alphabetical order), António Rito Silva, Antonio Vallecillo, Gustavo Rossi, João Araújo, João Falcão e Cunha, José Berrocal, José Corrales, José García-Fanjul, João Miguel Fernandes, Lidia Fuentes and María José Suárez-Cabal.

Finally, a special word of thanks to the sponsors of this conference, without whose contribution the event would have been somewhat less charming (not to mention gastronomically less satisfying).

Ana Moreira
Program Committee Chair

Prefácio

Celebrando 13 Anos de JISBD

Com a edição de 2008 em Gijón (7-10 Outubro), a Conferência em Engenharia de Software e Bases de Dados (JISBD) celebra 13 anos de existência. Apesar de ter nascido como um fórum onde a comunidade espanhola publicava os seus trabalhos e se reunia para discutir potenciais colaborações futuras de investigação, e até avaliar o estado de andamento dos projectos de investigação financiados pelo Ministério da Ciência e Tecnologia espanhola, há muito que extravasou essas fronteiras e cruzou oceanos.

Actualmente o JISBD é um marco importante na investigação dos mais jovens, mas também um fórum que os mais seniores não querem perder. Nos últimos anos a conferência abriu-se para o mundo inteiro, aceitando artigos escritos em Inglês, Espanhol e Português. Esta viragem trouxe não só mais participantes à conferência, mas também um aumento significativo do número de trabalhos submetidos e, principalmente, um aumento na qualidade desses trabalhos.

A comunidade do JISBD é agora auto-sustentada e em contínua expansão. A qualidade dos trabalhos aceites é equiparada à de muitos outros eventos internacionais de relevo. Por este motivo, nos últimos anos, foi-nos possível editar um volume especial no IEEE LA com uma versão estendida dos melhores trabalhos da conferência, o que acontecerá também nesta edição. Este volume, em conjunto com as actas formais da conferência com ISBN, é uma mostra da qualidade do trabalho que aqui se discute.

Uma das características de excelência desta conferência tem sido, desde sempre, o gabarito dos seus palestrantes convidados. É um prazer ver que muitos dos mais admirados investigadores e profissionais internacionais já foram convidados a falar para os participantes do JISBD.

Neste enquadramento fecundo para divulgação científica e tecnológica, a conferência inclui vários eventos satélite. Além dos artigos seleccionados para apresentação na conferência, o programa inclui ainda tutoriais, demonstrações de ferramentas, workshops para discussão de ideias inovadores e trabalhos em andamento, assim como um evento para a disseminação de trabalho de investigação já publicado em revistas e actas de conferências de grande prestígio (onde o índice de aceitação é inferior a 25% e o factor de impacto superior a 0.5).

Assim, não é excessivo afirmar que o JISBD se tem vindo a consolidar como um evento de referência onde investigadores e profissionais em Engenharia de Software e Bases de Dados se encontram para discutir, disseminar e trocar ideias, partilhar experiências e resultados entre diversos sectores e grupos de investigação, num contexto de excelente organização e invulgar hospitalidade.

Sobre esta Edição

A atestar o crescimento e internacionalização do JISBD está a origem dos artigos que nos chegaram. Este ano, a nacionalidade dos autores foi surpreendentemente diversificada, pois para além dois países Ibéricos e de dez países Latino-Americanos, recebemos trabalhos também da Alemanha, China, França, Índia, Irão e Paquistão.

O número total de resumos foi de 115, sendo que destes, 112 artigos foram submetidos para avaliação. Cada artigo foi avaliado por pelo menos três revisores, sendo que vários foram avaliados por quatro. O Comité de Programa aceitou 30 artigos longos e escolheu 12 para apresentação como artigos curtos. Assim, o índice de aceitação de artigos longos foi de cerca de 25%.

Este sucesso acarreta responsabilidades acrescidas em garantir a independência de julgamentos e em fazer cumprir a ética e as normas internacionais. É por este motivo que, nos últimos anos, se

tem feito um esforço muito grande para evitar submissões duplicadas, tarefa nem sempre fácil para os membros do Comitê de Programa, já que a conferência aceita três línguas de escrita. Este ano foram rejeitados três bons artigos avaliados como de submissão duplicada, em duas línguas, para eventos diferentes.

Para além dos artigos seleccionados, a conferência conta também com a organização de cinco *workshops*, um *tutorial*, nove demonstrações de ferramentas, um painel industrial e ainda um fórum onde se discutem trabalhos de relevo já publicados em revistas ou outras conferências.

Mas sem dúvida que os momentos mais altos da conferência são sempre marcados pelo admirável conjunto de palestrantes convidados. Este ano tivemos a sorte de receber Bashar Nuseibeh e de Bran Selic.

Bashar Nuseibeh é um académico e investigador da Open University, na Inglaterra, e professor convidado em várias outras universidades, incluindo o Instituto Japonês de Informática. Bashar preside vários comités internacionais e é admirado também pelo seu trabalho para a indústria, que inclui organizações como o National Air Traffic Services (NATS) do Reino Unido, Texas Instruments, Praxis Critical Systems, Philips Research Labs, e a NASA.

Bran Selic foi durante umas dezenas de anos engenheiro e investigador distinguido da IBM e actualmente preside uma empresa de consultoria internacional sediada no Canadá. É conhecido mundialmente pelos seus trabalhos em sistemas de larga escala industrial e também pelo seu pioneirismo nas áreas de desenvolvimento orientado a modelos e sistemas embutidos de tempo real.

A palestra do Bashar é intitulada “*The five W’s (and one “H”) of Security: ... Software Engineering of Secure Systems*”, enquanto que a do Bran é sobre “*Model-Based Software Engineering: Expected and Unexpected Challenges*”.

Agradecimentos

Uma palavra especial de agradecimento ao Bashar e ao Bran por terem aceite o meu convite e por brindarem todos os participantes com a sua experiência, conhecimento e refinado sentido de humor. Espero que o JISBD tenha sido também para eles uma experiência agradável e diferente.

Agradecimentos são justamente devidos ao grande número de colaboradores, sem o contributo dos quais, a conferência não poderia ter tido êxito. Aos autores, claro, por confiarem na qualidade do JISBD e submeterem, por isso, os seus trabalhos. Aos membros do Comitê de Programa cujas revisões asseguram que essa confiança continua a justificar-se.

Para gerir o sistema de submissão, contei com o apoio incondicional do Juan Hernández e do José Javier Berrocal. Eles foram os meus “anjos da guarda”, sempre atentos a todos os prazos e prontos a dar todas as explicações. Um agradecimento particular ao contributo meu “Comitê Executivo de Programa”, Antonio Vallecillo, Juan Hernández, Miguel Toro, Vicente Pelechano e Xavier Franch. Obrigada pelo vosso apoio e sugestões.

Obrigada aos organizadores principais da conferência, em especial ao Javier Tuya, e ao seu vice-presidente, Claudio de la Riva, pela gestão eficaz das inúmeras tarefas que uma conferência desta dimensão exige. Um agradecimento é ainda devido, e por ordem alfabética, aos responsáveis dos eventos satélite, António Rito Silva, Antonio Vallecillo, Gustavo Rossi, João Araújo, João Falcão e Cunha, José Berrocal, José Corrales, José García-Fanjul, João Miguel Fernandes, Lidia Fuentes e María José Suárez-Cabal.

Finalmente, um agradecimento aos patrocinadores da conferência, sem o contributo de quem o evento teria tido menos charme (e uma gastronomia muito menos requintada).

Ana Moreira
Presidente do Comitê de Programa

Prefacio

Con esta edición 2008 en Gijón (7 al 10 de Octubre), las Jornadas de Ingeniería del Software y Bases de Datos (JISBD) celebra 13 años de existencia. JISBD nació como un foro donde la comunidad española publicaba su trabajo, discutía potenciales colaboraciones en investigación y evaluaba el progreso de los proyectos de investigación financiados por el Ministerio de Ciencia y Tecnología, y en la actualidad ha traspasado fronteras y cruzado varios océanos.

Actualmente, la conferencia es una referencia importante para jóvenes investigadores, así como un foro de cita obligada para investigadores más experimentados. Durante los últimos años, JISBD se ha abierto al mundo, aceptando artículos en Inglés y Portugués, además de Castellano. Este cambio no solamente se ha traducido en más participantes, sino que ha incrementado significativamente el número de artículos enviados, y principalmente, la calidad de los artículos aceptados.

La comunidad JISBD está actualmente auto sustentada y continúa expandiéndose. La calidad de los trabajos aceptados es equivalente al de otros eventos internacionales relevantes. Durante los últimos años, ha sido posible editar un volumen especial de IEEE LA con versiones ampliadas de los mejores trabajos presentados en la conferencia, lo que sucederá también en la presente edición. Este volumen especial, junto con las actas de la conferencia con ISBN, es una muestra de la calidad de los trabajos de JISBD.

Una de las características más sobresalientes de la conferencia ha sido la calidad de los ponentes invitados. Varios investigadores y profesionales de reconocido prestigio internacional han sido invitados a participar como ponentes en JISBD.

Dentro de este marco científico y tecnológico, la conferencia incluye varios eventos relacionados. Además de la presentación de artículos originales de alta calidad en la conferencia principal, el programa incluye tutoriales, demostraciones de herramientas, talleres para la discusión de ideas innovadoras y trabajos en curso, así como la divulgación de trabajos de investigación publicados en revistas y conferencias de prestigio (con un ratio de aceptación por debajo del 25% y un factor de impacto por encima de 0,5).

No es una exageración afirmar que JISBD ha consolidado su posición como un evento de referencia donde investigadores y profesionales de la Ingeniería del Software y las Bases de Datos se reúnen para discutir resultados y compartir ideas. JISBD se ha convertido en un foro importante para la colaboración entre diferentes sectores y grupos de investigación, en un contexto de excelente organización y excepcional hospitalidad.

Sobre la presente edición

El crecimiento e internacionalización de JISBD se hace evidente analizando el origen de los artículos recibidos. En la presente edición, además de los artículos recibidos de los dos países de la Península Ibérica y los diez países Latinoamericanos, se han recibido artículos de China, Francia, Alemania, India, Irán y Pakistán.

De un total de 115 resúmenes previamente recibidos, finalmente se recibieron 112 artículos para su revisión. La mayoría de los artículos fueron revisados por tres miembros del Comité de Programa y varios por cuatro. El Comité de Programa aceptó 30 artículos largos y seleccionó 12 para su presentación como artículos cortos. El ratio de aceptación para los artículos largos fue de aproximadamente el 25%.

El éxito de la conferencia implica grandes responsabilidades en términos de garantizar la independencia de las revisiones y el cumplimiento de los estándares internacionales de ética. Por esta razón, durante los últimos años se ha realizado un mayor esfuerzo en aras de evitar envíos duplicados, una tarea especialmente dificultosa, ya que la conferencia acepta envíos en tres idiomas. En la

presente edición tres artículos fueron rechazados debido al doble envío en diferentes idiomas para diferentes eventos.

Además de los artículos aceptados, la conferencia incluye cinco talleres, un tutorial, nueve demostraciones de herramientas y foro para la discusión y divulgación de trabajos relevantes previamente publicados, así como una mesa redonda de carácter industrial.

Una característica importante de la conferencia es, sin ninguna duda, la excelencia de los ponentes invitados. La presente edición no es una excepción y estamos orgullosos de contar con la presencia de Bashar Nuseibeh y Bran Selic.

Bashar Nuseibeh es académico e investigador en la Open University del Reino Unido y profesor invitado en otras muchas universidades, incluyendo el Instituto Nacional Japonés de Informática. Bashar preside varios comités internacionales y está reconocido igualmente por su trabajo industrial, incluyendo organizaciones tales como el Servicio Nacional de Tráfico Aéreo del Reino Unido (NATS), Texas Instruments, Praxis Critical Systems, Philips Research Labs y la NASA.

Bran Selic fué durante varios años un destacado ingeniero e investigador en IBM y actualmente lidera una consultora internacional con sede en Canadá. Es internacionalmente conocido por su trabajo en sistemas industriales a gran escala y por su trabajo pionero en Desarrollo Dirigido por Modelos y Sistemas Empotrados en Tiempo Real.

La conferencia de Bashar se titula *“The five W’s (and one “H”) of Security: ... Software Engineering of Secure Systems”* y la de Bran *“Model-Based Software Engineering: Expected and Unexpected Challenges”*.

Agradecimientos

Un agradecimiento especial es para Bashar y Bran por haber aceptado mi invitación y por compartir con todos los participantes sus conocimientos, experiencia y refinado sentido del humor.

Agradecimientos también para la multitud de colaboradores sin los cuales el éxito de la conferencia no habría sido posible. En primer lugar, para los autores de los artículos por confiar en la calidad de JISBD y enviar sus trabajos. En segundo lugar, para los miembros del Comité de Programa, cuyas revisiones aseguran la calidad de los trabajos.

Para el proceso de gestión y revisión de los trabajos recibidos, fui afortunada por tener la ayuda constante de Juan Hernández y José Javier Berrocal. Ellos fueron mis ángeles guardianes, alertándome constantemente de las fechas límite y siempre preparados para ayudarme cuando lo necesitaba. Agradecimientos especiales por la contribución de mi “Comité de Programa Ejecutivo”, Antonio Vallecillo, Juan Hernández, Miguel Toro, Vicente Pelechano y Xavier Franch.

Agradecimientos también para los organizadores de la conferencia principal, especialmente al presidente del comité organizador Javier Tuya y su vicepresidente Claudio de la Riva, por su manejo eficiente de las numerosas tareas que una conferencia de este tamaño y calidad conllevan. Agradecimientos también para los responsables de los eventos relacionados (en orden alfabético) António Rito Silva, Antonio Vallecillo, Gustavo Rossi, João Araújo, João Falcão e Cunha, José Berrocal, José Corrales, José García-Fanjul, João Miguel Fernandes, Lidia Fuentes y María José Suárez-Cabal.

Finalmente, palabras especiales de agradecimiento para los patrocinadores de la conferencia, sin cuya contribución el evento habría sido menos encantador (y con una gastronomía menos refinada).

Ana Moreira
Presidenta del Comité de Programa

Conference Committee

Program Committee Chair

Ana Moreira (Univ. Nova de Lisboa, Portugal)

Organizing Chair

Javier Tuya (Univ. Oviedo, Spain)

Organizing Co-Chair

Claudio de la Riva (Univ. Oviedo, Spain)

Permanent Committee Secretary

Mario Piattini (Univ. Castilla-La Mancha, Spain)

Tutorial Chair

António Rito Silva (Univ. Técnica Lisboa, Portugal)

Workshop Chair

João Araújo (Univ. Nova de Lisboa, Portugal)

Tool Demonstrations Chair

Lidia Fuentes (Univ. Málaga, Spain)

Relevant Papers Dissemination Chairs

Antonio Vallecillo (Univ. Málaga, Spain)

João Falcão Cunha (Univ. Porto, Portugal)

Proceedings Chair

María José Suárez-Cabal (Univ. Oviedo, Spain)

Cyber Chair

Jose Javier Berrocal (Univ. Extremadura, Spain)

Web Chair

José A. Corrales (Univ. Oviedo, Spain)

Publicity Chairs

Gustavo Rossi (Univ. La Plata, Argentina)

José García-Fanjul (Univ. Oviedo, Spain)

João Miguel Fernandes (Univ. Minho, Portugal)

Organizing Committee (Univ. Oviedo, Spain)

Javier Tuya
Claudio de la Riva
José García-Fanjul
Isabel Sevilla
María José Suárez-Cabal
José Ramón de Diego
Raquel Blanco
Eugenia Díaz Fernández
José A. Corrales
Marta Fernández de Arriba

SISTEDES Executive Board

President

Miguel Toro (Univ. Sevilla, Spain)

Vice President

Juan José Moreno (Univ. Polit. Madrid, Spain)

Secretary

Nieves R. Brisaboa (Univ. Coruña, Spain)

Treasurer

Javier Tuya (Univ. Oviedo, Spain)

Members

Pere Botella (Univ. Polit. Catalunya, Spain)
Ricardo Peña (Univ. Complutense Madrid, Spain)
Coral Calero (Univ. Castilla-La Mancha, Spain)
Manuel Hermenegildo (Univ. Polit. Madrid, Spain)
Ernesto Pimentel (Univ. Málaga, Spain)
María Ribera Sancho (Univ. Polit. Catalunya, Spain)
Natalia Juristo (Univ. Polit. Madrid, Spain)
Salvador Lucas (Univ. Polit. Valencia, Spain)

Submission and Review Support System (Quercus Software Engineering Group)

Javier Berrocal (Univ. Extremadura, Spain)
Juan Hernández (Univ. Extremadura, Spain)

Secretariat

Fundación Universidad de Oviedo
C/ Principado 3, 4ª Planta
33007 Oviedo, Spain.
Tel: 34-985104927
Fax: 34-985104928

Executive Program Committee

Xavier Franch (Univ. Polit. Catalunya, Spain)
Juan Hernández (Univ. Extremadura, Spain)
Vicente Pelechano (Univ. Polit. Valencia, Spain)
Antonio Vallecillo (Univ. Málaga, Spain)
Miguel Toro (Univ. Sevilla, Spain)
Javier Tuya (Univ. Oviedo, Spain)

Program Committee

Albert Abelló (Univ. Polit. Catalunya, Spain)
Ana Paula Afonso (Univ. Lisboa, Portugal)
Ademar Aguiar (Univ. Porto, Portugal)
Jesús Aguilar (Univ. Sevilla, Spain)
José Aldana (Univ. Málaga, Spain)
Mauricio Alférez (U. Nova de Lisboa, Portugal)
Bárbara Álvarez (Univ. Polit. Cartagena, Spain)
Raquel Anaya (Univ. EAFIT, Colombia)
María José Aramburu (Univ. Jaume I, Spain)
Hernán Astudillo (U. T. Federico Santa María, Chile)
Orlando Belo (Univ. do Minho, Portugal)
Rafael Berlanga (Univ. Jaume I, Spain)
Paulo Borba (Univ. Federal Pernambuco, Brazil)
Pere Botella (Univ. Polit. Catalunya, Spain)
Rosana Braga (Univ. São Paulo, Brazil)
Nieves Brisaboa (Univ. Coruña, Spain)
Isabel Brito (Inst. Polit. Beja, Portugal)
Fernando Brito e Abreu (U. Nova de Lisboa, Portugal)
Coral Calero (Univ. Castilla-La Mancha, Spain)
Marcelo Campo (UNICEN, Argentina)
Carlos Canal (Univ. Málaga, Spain)
Valeria de Castro (Univ. Rey Juan Carlos, Spain)
Matilde Celma (Univ. Polit. Valencia, Spain)
Christina Chávez (Univ. Bahia, Brazil)
Rafael Corchuelo (Univ. Sevilla, Spain)
Dolors Costal (Univ. Polit. Catalunya, Spain)
Yania Crespo (Univ. Valladolid, Spain)
Carlos Delgado (Univ. Carlos III, Spain)
Oscar Díaz (Univ. País Vasco, Spain)
Javier Dolado (Univ. País Vasco, Spain)
Xavier Franch (Univ. Polit. Catalunya, Spain)
Pablo de la Fuente (Univ. Valladolid, Spain)
Mario Gaspar da Silva (Univ. Lisboa, Portugal)
Alessandro García (Univ. Lancaster, UK)
Marcela Genero (Univ. Castilla-La Mancha, Spain)
Cristina Gómez (Univ. Polit. Catalunya, Spain)
Jaime Gómez (Univ. Alicante, Spain)
Alfredo Goñi (Univ. País Vasco, Spain)
Silvia Gordillo (UNLP, Argentina)
Pedro Guerreiro (Univ. Algarbe, Portugal)
Juan Hernández (Univ. Extremadura, Spain)
Jon Iturrioz (Univ. País Vasco, Spain)
Elena Jurado (Univ. Extremadura, Spain)
Natalia Juristo (Univ. Polit. Madrid, Spain)
Miguel Katrib (Grupo WEBOO, Cuba)
María Lencastre (Univ. Pernambuco, Brazil)
Antonia Lopes (Univ. Lisboa, Portugal)
Adolfo Lozano (Univ. Extremadura, Spain)
Esperanza Marcos (Univ. Rey Juan Carlos, Spain)
Henrique Madeira (Univ. Coimbra, Portugal)
Eduardo Mena (Univ. Zaragoza, Spain)
Ana María Moreno (Univ. Polit. Madrid, Spain)
Juan José Moreno (Univ. Polit. Madrid, Spain)
Juan Manuel Murillo (Univ. Extremadura, Spain)
Oscar Pastor (Univ. Polit. Valencia, Spain)
Vicente Pelechano (Univ. Polit. Valencia, Spain)
Marcelo Pimenta (Univ. F. Rio Grande do Sul, Brazil)
Ernesto Pimentel (Univ. Málaga, Spain)
Mónica Pinto (Univ. Málaga, Spain)
Ángeles Places (Univ. Coruña, Spain)
Antonio Polo (Univ. Extremadura, Spain)
Claudia Pons (UNICEN, Argentina)
Tom Price (Univ. F. Rio Grande do Sul, Brazil)
Carme Quer (Univ. Polit. Catalunya, Spain)
Celia Ramos (Univ. Algarbe, Portugal)
Isabel Ramos (Univ. Sevilla, Spain)
Isidro Ramos (Univ. Polit. Valencia, Spain)
Claudio de la Riva (Univ. Oviedo, Spain)
José Riquelme (Univ. Sevilla, Spain)
José Luis Roda (Univ. La Laguna, Spain)
María José Rodríguez Fortis (Univ. Granada, Spain)
José Raúl Romero (Univ. Córdoba, Spain)
Antonio Ruiz (Univ. Sevilla, Spain)
Francisco Ruiz (Univ. Castilla-La Mancha, Spain)
José Samos (Univ. Granada, Spain)
Fernando Sánchez (Univ. Extremadura, Spain)
Juan Sánchez (Univ. Polit. Valencia, Spain)
Carla Silva (Univ. F. Pernambuco, Brazil)
Ernest Teniente (Univ. Polit. Catalunya, Spain)
Miguel Toro (Univ. Sevilla, Spain)
Ambrosio Toval (Univ. Murcia, Spain)
Juan Carlos Trujillo (Univ. Alicante, Spain)
Toni Urpi (Univ. Polit. Catalunya, Spain)
Antonio Vallecillo (Univ. Málaga, Spain)
Belén Vela (Univ. Rey Juan Carlos, Spain)

Referees

Álvaro E. Prieto Ramos
Amador Durán Toro
André L. Santos
Andrea Delgado
Ángel Herranz
Angélica Caro
Anna Grimán Padua
Antonio Jesús Roa Valverde
Antônio Oliveira Filho
Antonio Ruiz-Cortés
Arturo Zambrano
Carlos Bobed
Carlos D. Barranco González
Carlos Enrique Cuesta Quintero
Carlos Neil
Cecilia Delgado Negrete
César J. Acuña
Claudio Sant' Anna
Cristina Vicente Chicote
Daniel Rodríguez
Dante Carrizo
Diana Marcela Sánchez
Diego Alonso Cáceres
Diego Seco Naveiras
Domingo Savio Rodríguez Baena
Eduardo Rodríguez López
Elisa Yumi Nakagawa
Ellen Francine Barbosa
Encarna Sosa Sánchez
Fernando Molina Molina
Fran J. Ruiz Bertol
Francisco Javier Lucas Martínez
Francisco Luís Gutiérrez Vela
Francisco Martínez Álvarez
Ignacio García Rodríguez de Guzmán
Ismael Caballero
Ismael Navas Delgado
Ismael Sanz Blasco
Javier Pérez García
Joaquín Lasheras
Joaquín Nicolás
Jorge Gracia
Jorge Martínez Gil
José María Cavero Barca
Juan Ángel Pastor Franco
Juan M. Vara
Juan Manuel Pérez Martínez
Manuel Ángel Serrano Martín
Manuel Resinas
Márcio de Medeiros Ribeiro
Marcirio Chaves
Marcos López Sanz
Mari Carmen Otero
María Esperanza Manso Martínez
María Luisa Rodríguez Almendros
María Teresa Gómez López
María Visitación Hurtado Torres
Martin Solari
Miguel Ángel Laguna Serrano
Miguel Ángel Martínez
Miguel Rodríguez Luaces
M^a Ángeles Moraga de la Rubia
Nuno Cardoso
Orlando Avila-García
Oscar Dieste
Óscar Pedreira Fernández
Othmane Chniber
Pablo Inostroza
Pablo Trinidad
Paloma Cáceres García de Marina
Pedro Sánchez Palma
Raquel M. Crespo García
Raquel Trillo Lado
Roberto Almeida Bittencourt
Roberto Rodríguez Echeverría
Roberto Ruiz
Rui Lopes
Sascha Ossowski
Sergio Ilarri Artigas
Vicente Luque Centeno

Sponsors



Ayuntamiento de Gijón



GOBIERNO DEL
PRINCIPADO DE ASTURIAS



Table of Contents¹

Keynote Address 1

The five W's (and one "H") of Security: Software Engineering of Secure Systems	1
<i>Bashar Nuseibeh</i>	

Aspects

Analysis of Modularity by an Aspect-Oriented Measurement Process.....	3
<i>José Conejero, Juan Hernández, Elena Jurado, Klaas Berg</i>	

Process Engineering

Automating the Software Process Management.....	15
<i>Javier Berrocal, José Manuel García, Juan Manuel Murillo</i>	

Software Product Lines

Generación Automática de Casos de Prueba en Líneas de Producto	27
<i>Pedro Mateo, Beatriz Lamanha, Macario Usaola</i>	
Gestión de la Variabilidad de los Requisitos de Seguridad en Líneas de Producto	39
<i>Daniel Mellado, Eduardo Fernandez-Medina, Mario Piattini</i>	

¹ The section headings below correspond to the conference program, but do not include all the presentations in each conference session (where short papers and dissemination papers on the same topic also were included). Thus, the sections here all contain fewer papers than the corresponding conference session; the short papers are listed separated in this volume, followed by a chapter with an overview of the dissemination papers.

Information Engineering

Clasificación de Imágenes en el Sistema Qatris Imanager Mediante Regresión Logística Bayesiana	51
<i>Inés Horrillo, Manuel Barrena</i>	
Efficient Retrieval of Ontology Fragments Using an Interval Labeling Écheme ...	63
<i>Victoria Romero, Rafael Llavori</i>	
Un Modelo para el Análisis y Explotación de Información Cognitiva en Repositorios Documentales	75
<i>Miguel A. Martínez-Prieto, Joaquín Adiego, Pablo de la Fuente</i>	
Un Sistema de Consulta sobre Documentos Transformados con LZCS.....	87
<i>Joaquín Adiego, Gonzalo Navarro, Pablo de la Fuente</i>	

Model Engineering

Análisis de Series Temporales Dirigido por Modelos Conceptuales sobre Datos Multidimensionales.....	99
<i>Jose Zubcoff, Jesús Pardillo, Juan Trujillo</i>	
Una Aproximación Dirigida por Modelos para el Desarrollo de Esquemas XML.....	111
<i>Verónica Bollati, Juan Vara, Belén Vela, Esperanza Marcos</i>	
Generación de Metadatos OLAP Dirigida por Modelos sobre Almacenes de Datos	123
<i>Juan Trujillo, Jesús Pardillo, Jose-Norberto Mazón</i>	

Formal Methods

Modelling Mash-up Resources	135
<i>Iván Pérez, Ángel Herranz, Susana Muñoz, Juan Moreno-Navarro</i>	
Optimizando el Funcionamiento del Algoritmo FOIL	147
<i>Pablo Palacios, José Arjona, José Álvarez, Iñaki Fernández de Viana</i>	
Towards the Correctness Verification of Business Processes Modelled with UML.....	159
<i>Luis Mendoza, Manuel Capel, Kawtar Akhlaki</i>	

Maintenance and Testing

Agil_MANTEMA: Una Metodología de Mantenimiento de Software para Pequeñas Organizaciones	171
<i>Francisco Pino, Francisco Ruiz, Jorge Triñanes, Félix García, Mario Piattini</i>	
Priorización del Valor de Artefactos Software Basada en la Frecuencia de Uso..	183
<i>Daniel Cabrero, Javier Garzas, Mario Piattini</i>	
Identificación de Fallos en Módulos Software	195
<i>José Riquelme, Roberto Ruiz, Daniel Rodríguez</i>	

Data Mining, Data Streaming and Datawarehouses

Hacia la Implementación Automática de Almacenes de Datos Seguros en Herramientas OLAP.....	205
<i>Carlos Blanco, Ignacio García-Rodríguez de Guzmán, Eduardo Fernández-Medina, Juan Trujillo, Mario Piattini</i>	
Una aproximación Basada en Diagramas de Actividades de UML para el Modelado Conceptual de Procesos ETL en Almacenes de Datos.....	217
<i>Lilia Muñoz, Jose-Norberto Mazón, Jesús Pardillo, Juan Trujillo</i>	
MeCADI*: un Marco Orientado a Objetivos para el Modelado de la Calidad en Almacenes de Datos.....	229
<i>Cristina Cachero, Jesús Pardillo, Jose-Norberto Mazón, Juan Trujillo</i>	

Reengineering and Software Modernization

Reverse Engineering of Object-Relational Database Schemas	241
<i>Jordi Cabot, Cristina Gómez, Elena Planas, M. Elena Rodríguez</i>	

Quality, Measurement & Estimation of Products & Processes

Una Metodología Basada en ISO/IEC 15939 para la Elaboración de Planes de Medición de Calidad de Datos.....	253
<i>Eugenio Verbo, Ismael Caballero, Ricardo Pérez, Coral Calero, Mario Piattini</i>	
Metodologías para Definir Programas de Medición en PyMEs: El Marco MIS-PyME.....	265
<i>María Díaz-Ley, Félix García, Mario Piattini</i>	

Visualización de la Usabilidad de Componentes Software.....	275
<i>M^a Ángeles Moraga, Sergio Susín, Virginia Arcos, Coral Calero</i>	
Aportaciones de una Visualización Metafórica al Análisis de Proyectos Software	287
<i>Amaia Aguirregoitia, J.Javier Dolado</i>	
Aplicación de las Técnicas de Modelado y Simulación en la Gestión de la Capacidad de los Servicios TI.....	299
<i>Elena Orta Cuevas, Mercedes Ruiz Carreira, Miguel Toro Bonilla</i>	
Measure Assessment for Heterogeneous XML Collections.....	311
<i>María Pérez Catalán, Ismael Sanz, Rafael Berlanga</i>	

Requirements Engineering

Revisiones Sistemáticas: Recomendaciones para un Proceso Adecuado a la Ingeniería del Software	321
<i>Oscar Dieste, Anna Grimán, Marta López</i>	
Metodologías Ágiles desde la Perspectiva de la Especificación de Requisitos Funcionales y No-Funcionales	333
<i>Pilar Rodríguez, Agustín Yagüe, Pedro Alarcón, Juan Garbajosa</i>	
Metamodelo y Perfil UML para el Modelado Orientado a Metas de Requisitos Medibles.....	345
<i>Fernando Molina, Cristina Cachero, Jesús Pardillo, Ambrosio Toval</i>	

Keynote Address 2

Model-Based Software Engineering: Expected and Unexpected Challenges.....	357
<i>Bran Selic</i>	

Short Papers

AAJ: Un Lenguaje de Descripción Arquitectónica Orientado a Aspectos.....	361
<i>María Boton, Amparo Navasa</i>	
An Ontology for IT Services	367
<i>Jorge Freitas, Anacleto Correia, Fernando Abreu</i>	

Construcción de Modelos Lógicos Multidimensionales Seguros para su Implementación en Herramientas OLAP Mediante MDA y QVT	373
<i>Carlos Blanco, Ignacio García-Rodríguez de Guzmán, Eduardo Fernández-Medina, Juan Trujillo, Mario Piattini</i>	
Desarrollo de Almacenes de Datos Espacio Temporales Dirigido por Modelos ..	379
<i>Octavio Glorio, Juan Trujillo</i>	
Generating Domain Specific Aspect Code for Navigation from Platform Specific Models in MWACSL.....	385
<i>Antonia M. Reina Quintero, Miguel Toro Bonilla, Jesús Torres Valderrama</i>	
Zentipede: Una Contribución a la Renovación de la Gestión del Proceso Software	391
<i>José Manuel García Alonso, José Javier Berrocal, Juan Manuel Murillo Rodríguez</i>	
Hacia la Definición de un Simulador para la Enseñanza de la Elicitación de Requisitos en el Contexto del Desarrollo Global del Software	417
<i>Miguel Romero, Aurora Vizcaino, Mario Piattini</i>	
Un Marco de Referencia para Comparar ESBs desde la Perspectiva de la Integración de Aplicaciones.....	403
<i>Rafael Corchuelo, Rafael Frantz, Jesús González</i>	
Refactorizaciones en la Migración del Software.....	409
<i>Rául Marticorena, Yania Crespo, Carlos López</i>	
Diseño Evolutivo de Bases de Datos XML	415
<i>Carlos Nilo, Cecilia Reyes, Jose Marti</i>	
Impacto de las Multiplicidades en la Resolución de Problemas de Sumarizabilidad para OLAP	421
<i>Jose-Norberto Mazón, Jens Lechtenbörger, Juan Trujillo</i>	

Workshops, Tutorials, Demos and Dissemination

Workshops.....	427
<i>João Araújo</i>	
Tutorials	429
<i>António Rito Silva</i>	

Tool Demonstrations	431
<i>Lidia Fuentes</i>	
ActiveRulesDBX – Ferramenta para Execução de Regras a partir da Detecção de Eventos Temporais.....	433
<i>Eugênio de O. Simonetto, Jéferson Kasper, Giovanni R. Librelotto</i>	
Deriving AO Software Architectures using the AO-ADL Tool Suite	437
<i>Mónica Pinto, Lidia Fuentes, Luis Fernández, Juan A. Valenzuela</i>	
ESFORA: a tool for the dEfinition of domain SPecific OpeRation languages.....	441
<i>David Musat, Jennifer Pérez, Pedro P. Alarcón, Agustín Yagüe</i>	
FAMA Framework	445
<i>Pablo Trinidad, David Benavides, Antonio Ruiz-Cortés, Sergio Segura</i>	
ProSÉ: A Protégé plugin for Reusing Ontologies, Safe and Économique	449
<i>Ernesto Jiménez-Ruiz, Bernardo Cuenca Grau, Ulrike Sattler Thomas Schneider, Rafael Berlanga</i>	
REMM-Studio+: Extensiones para Modelar Variabilidad y Permitir la Reutilización de Requisitos	453
<i>Begoña Moros, Cristina Vicente-Chicote, Ambrosio Toval</i>	
RUX-Tool: Una herramienta CASE para el modelado y la generación automática de Interfaces de Usuario para RIA	457
<i>Marino Linaje, Juan Carlos Preciado, Fernando Sánchez-Figueroa Rober Morales-Chaparro, David Gordillo, Fernando Sánchez-Herrera</i>	
StateML+: Diseño, Validación y Generación de Código Ada para Máquinas de Estado Jerárquicas	461
<i>Diego Alonso, Cristina Vicente-Chicote, Bárbara Álvarez</i>	
Relevant Papers Dissemination	465
<i>Antonio Vallecillo, João Falcão Cunha</i>	
Feature Oriented Model Driven Development: A Case Study for Portlets.....	467
<i>Salvador Trujillo, Don Batory, Oscar Díaz</i>	
DEX: High-Performance Exploration on Large Graphs for Information Retrieval.....	69
<i>Norbert Martínez-Bazan, Victor Muntés-Mulero, Sergio Gómez-Villamor, Jordi Nin, Mario-A. Sánchez-Martínez, Josep-L. Larriba-Pey</i>	
Determining Criteria for Selecting Software Components: Lessons Learned	471
<i>Juan Pablo Carvallo, Xavier Franch, Carme Quer</i>	

Engineering Rich Internet Application User Interfaces over Legacy Web Models	473
<i>Marino Linaje, Juan Carlos Preciado, Fernando Sánchez-Figueroa</i>	
Guideliness for Eliciting Usability Functionalities	475
<i>Natalia Juristo, Ana María Moreno, Maria-Isabel Sánchez-Segura</i>	
From Wrapping to Knowledge	477
<i>José Luis Arjona, Rafael Corchuelo, David Ruiz, Miguel Toro</i>	
Introducing Structure Management in Automatic Reference Resolution: An XML-based Approach	479
<i>M. Mercedes Martínez-González, Pablo de la Fuente</i>	
Run-time Composition and Adaptation of Mismatching Behavioural Transactions	481
<i>Javier Cámara, Gwen Salaün, Carlos Canal</i>	
Building Domain-Specific Languages for Model-Driven Development	483
<i>Jesús Sánchez Cuadrado, Jesús García Molina</i>	
Reconciling requirement-driven data warehouses with data sources via multidimensional normal forms.....	485
<i>Jose-Norberto Mazón, Juan Trujillo, Jens Lechtenbörger</i>	
Developing Secure Data Warehouses with a UML Extension.....	487
<i>Eduardo Fernández-Medina, Juan Trujillo, Rodolfo Villarroel, Mario Piattini</i>	
Author Index.....	489

Agil_MANTEMA: Una metodología de mantenimiento de software para pequeñas organizaciones

Francisco J. Pino¹, Francisco Ruiz², Jorge Triñanes³, Félix García², Mario Piattini²

¹ Grupo IDIS

Facultad de Ingeniería Electrónica y Telecomunicaciones. Universidad del Cauca
Calle 5 No. 4 – 70. Popayán, Cauca, Colombia.
fjpino@unicauca.edu.co

² Grupo ALARCOS

Escuela Superior de Informática. Universidad Castilla-La Mancha
Paseo de la Universidad 4, Ciudad Real, España.
{Francisco.RuizG, Felix.Garcia, Mario.Piattini}@uclm.es

³ Facultad de Ingeniería, Universidad de la República
Julio Herrera y Reissig 565, Montevideo, Uruguay.
Triniane@fing.edu.uy

Resumen. El mantenimiento de software es una actividad muy importante y crítica para las empresas software. Además el mantenimiento de software es una oportunidad de negocio para muchas organizaciones, entre ellas las pequeñas empresas. En este sentido para ayudar a las empresas de este tipo que estén interesadas en fortalecer su proceso de mantenimiento, nosotros presentamos en este artículo a Agil_MANTEMA que es una propuesta metodológica de mantenimiento de software enfocada a pequeñas empresas. Esta metodología define un proceso ágil de mantenimiento detallando qué debe realizarse, cuándo, cómo y por quién, es decir, busca guiar paso a paso este proceso para este tipo de organizaciones. Agil_MANTEMA también define un conjunto de elementos como son tipos de mantenimiento, niveles de servicio y niveles de capacidad con el objetivo de: (i) manejar la complejidad inherente al proceso de mantenimiento y (ii) permitir a una pequeña empresa definir su propio proceso de mantenimiento teniendo en cuenta sus propias características y necesidades. Con el fin de mejorar su proceso de mantenimiento, dos pequeñas empresas condujeron un proyecto de mejora en el cual se utilizó Agil_MANTEMA satisfactoriamente como modelo que propone buenas y adecuadas prácticas para el mantenimiento de software.

Palabras Clave: Mantenimiento de software, Metodología MANTEMA, Pequeñas organizaciones software, Scrum, COMPETISOFT.

1 Introducción

Diferentes estudios señalan que el mantenimiento de software es la parte más costosa del ciclo de vida del software y que, en general, el porcentaje de recursos necesarios para mantenimiento se incrementa a medida que se produce más software. Concretamente en [5] se analiza el panorama de la industria latinoamericana del software, mostrando que cuanto mayor es la proporción entre los contratos de mantenimiento de software y la facturación total en productos de una empresa, menor es el número de nuevos clientes. La forma inversa también se presenta, es decir, cuanto menor es la proporción entre los contratos de mantenimiento y la facturación total de una empresa, mayor es la presencia de clientes nuevos en la cartera de la empresa. Éstos son algunos elementos que colocan en evidencia lo importante y crítico que es el mantenimiento del software para las empresas que conforman la industria del software.

Además es bien conocido que la industria del software está compuesta mayoritariamente por pequeñas organizaciones (empresas con menos de 50 empleados). Debido al modelo caótico de desarrollo de software de numerosas organizaciones de este tipo [2] y a las presiones del mercado, es habitual que dichas organizaciones inviertan mucho esfuerzo en actividades de mantenimiento de los productos software que han sido liberados, todo ello sin un adecuado aseguramiento de calidad. Además, algunas de estas organizaciones pretenden aprovechar el mantenimiento de software como una oportunidad de negocio, mediante su especialización en el mantenimiento de productos software desarrollados por otras organizaciones, es decir, mediante el ofrecimiento de un servicio de mantenimiento externalizado (outsourcing).

Considerando este panorama, es importante ofrecer a las pequeñas organizaciones métodos y guías que les permitan realizar y gestionar de una manera adecuada las actividades de mantenimiento de software. En esta línea, este artículo presenta una propuesta metodológica de mantenimiento de software enfocada a pequeñas organizaciones denominada Agil_MANTEMA, la cual define un proceso de mantenimiento detallando qué debe realizarse, cuándo, cómo y por quién, es decir, busca guiar paso a paso el proceso de mantenimiento del software en este tipo de organizaciones. De forma complementaria, también se presenta la utilización de esta metodología por dos empresas que actualmente están llevando a cabo un esfuerzo de mejora de procesos.

Agil_MANTEMA ha sido desarrollada en el contexto del proyecto iberoamericano COMPETISOFT [6], el cual busca incrementar el nivel de competitividad de las pequeñas organizaciones software de la región mediante la creación y difusión de un marco metodológico común para la mejora de procesos.

El resto del artículo se estructura de la siguiente manera: la sección 2 brinda una visión general de Agil_MANTEMA, la sección 3 describe el proceso de mantenimiento, y a continuación la sección 4 muestra la utilización de Agil_MANTEMA en un ciclo de mejora de procesos de dos pequeñas empresas desarrolladoras de software. Finalmente, se describen las principales conclusiones y el trabajo futuro.

2 Visión general de Agil_MANTEMA

Agil_MANTEMA está creado a partir de la agilización de MANTEMA [9] a través de la aplicación de la metodología Scrum [11] para la gestión ágil de proyectos.

MANTEMA fue desarrollada pensando en medianos y grandes proyectos y servicios de mantenimiento, especialmente externalizados, por ejemplo, para banca o administraciones públicas. Por ello, se caracterizó por seguir un proceso de mantenimiento basado en la norma ISO 12207 definido de forma muy elaborada: flujo de actividades y tareas a realizar, modelo de roles y organizaciones participantes, entradas y salidas de las actividades, relación de técnicas aconsejadas en cada actividad, y una revisión de los tipos de mantenimiento en base a la experiencia en proyectos medianos y grandes. En este último aspecto, se establecía una diferencia clara entre el flujo de trabajo a realizar en el mantenimiento correctivo urgente (no planificable) y el resto de tipos de mantenimiento (planificables) [9].

Por otra parte, algunos métodos ágiles como Agile modeling, Extreme Programming, Pragmatic programming y Scrum, se centran en pequeños equipos [1]. Sin embargo, de éstos métodos sólo Scrum presta apoyo a la gestión de proyectos (los demás describen prácticas y técnicas alineadas con el desarrollo de software). Como método, Scrum hace énfasis en valores y prácticas de gestión ágiles sin incluir ninguna técnica. Esto le permite a Scrum que sea válido para aportar a otros métodos y procesos una gestión o manera de funcionar más ágil. Así, el proceso incremental y evolutivo, propuesto por Scrum para la gestión de los proyectos o servicios, ha sido combinado, integrado y adaptado con MANTEMA con el fin de crear Agil_MANTEMA.

El modelo general de proceso de Agil_MANTEMA resultante de esta integración se presenta en la Fig. 1. Este modelo de proceso se complementa con los siguientes elementos que también han sido adaptados para integrarlos en Agil_MANTEMA: (i) Nivel de servicio, extraído de Métrica V3

[4]; y (ii) Nivel de capacidad del proceso, basado en la norma ISO/IEC 15504 [3]. Estos dos conceptos se introducen y relacionan en Agil_MANTEMA de forma ortogonal o bidimensional, de forma que en una dimensión se encuentran los niveles de servicio y en la otra dimensión se encuentran los niveles de capacidad del proceso de mantenimiento (ver Tabla 1).

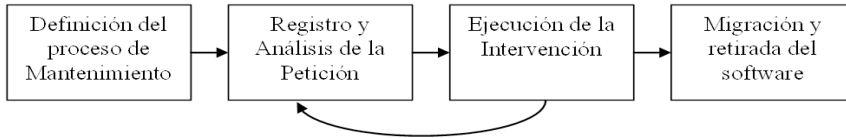


Fig. 1. Modelo de proceso general de Agil_MANTEMA

Tabla 1. Representación bidimensional de Agil_MANTEMA

		Niveles de Servicio		
		Avanzado	Intermedio	Básico
Niveles de Capacidad	Uno (N1)			
	Dos (N2)			
	Tres (N3)			

El propósito de ofrecer esta representación bidimensional es que cualquier pequeña organización pueda manejar mejor la complejidad inherente al proceso de mantenimiento de software, mediante una adaptación de este proceso en función de sus propias características organizacionales y de sus objetivos de negocio. La intención es que una organización pequeña pueda elegir el nivel de servicio que desea ofrecer en un proyecto o servicio concreto, y el nivel de capacidad que desea implementar en su proceso de mantenimiento, de acuerdo a sus necesidades e infraestructura. Como se observa en la Tabla 1, Agil_MANTEMA plantea que el nivel de servicio básico solo tiene un nivel de capacidad, mientras que el nivel de servicio intermedio puede realizarse con dos niveles de capacidad diferentes, y el nivel de servicio avanzado puede llevarse a cabo con tres niveles de capacidad distintos.

2.1 Niveles de servicio, niveles de capacidad y tipos de mantenimiento

Agil_MANTEMA define tres niveles de servicio e identifica los respectivos tipos de mantenimiento que se atienden en cada nivel (ver Tabla 2). Además, también define las interfaces que tiene cada nivel de servicio con los otros procesos que brindan apoyo al proceso de mantenimiento.

Tabla 2. Niveles de servicio de Agil_MANTEMA

	Básico	Intermedio	Avanzado
Tipos de Mantenimiento atendidos	- Correctivo Urgente	- Correctivo Urgente - Correctivo No Urgente - Perfectivo	- Correctivo Urgente - Correctivo No Urgente - Perfectivo - Adaptativo - Preventivo
Interfaces fundamentales con otros procesos	- Soporte al cliente (N1) - Gestión de resolución de problemas (N1)	- Soporte al cliente (N1) - Gestión de resolución de problemas (N1) - Gestión de la Configuración (N2) - Aseguramiento de la Calidad (N2)	- Soporte al cliente (N1) - Gestión de resolución de problemas (N1) - Gestión de la Configuración (N2) - Aseguramiento de la Calidad (N2) - Gestión de cambio de requisitos (N3) - Gestión de proyectos (N3)

Los tipos de mantenimiento en Agil_MANTEMA son los cinco identificados en MANTEMA [10], ya que éste no es un factor que se vea afectado por la búsqueda de mayor agilidad. Los tipos

de mantenimiento son organizados en las categorías planificable y no planificable. El mantenimiento correctivo urgente es no planificable, mientras que los mantenimientos correctivo no urgente, perfectivo, adaptativo y preventivo son planificables. La división del mantenimiento en estas dos categorías permite lograr una mejor gestión y optimización del grupo ordenado de peticiones de mantenimiento, ofreciendo un criterio para clasificar y priorizar las peticiones al responsable de llevar a cabo esta tarea.

Por otra parte, la implementación de las actividades descritas por las interfaces permite llevar a cabo prácticas base y de gestión que incrementan el nivel de capacidad del proceso de mantenimiento. Siguiendo esta estrategia, en Agil_MANTEMA se establecen tres niveles de capacidad para el proceso de mantenimiento, en función de las interfaces implementadas. Cada una de las interfaces se relaciona con un nivel de capacidad determinado (ver Tabla 2). Por ejemplo, se puede tener un proceso de mantenimiento con nivel de servicio intermedio y nivel de capacidad 1, lo cual indica que los tipos de mantenimiento que se realizan son correctivo urgente, no urgente y perfectivo; y además se implementan ampliamente las interfaces Soporte al cliente y Gestión de resolución de problemas. Para que este proceso de mantenimiento tenga un nivel de capacidad 2, las dos interfaces (de nivel 1) descritas anteriormente deben implementarse completamente y también se debe implementar ampliamente las interfaces de nivel 2: Gestión de la Configuración y Aseguramiento de la Calidad.

Dependiendo de las necesidades y características propias de cada organización, así como de las circunstancias particulares de cada proyecto o servicio de mantenimiento concreto, se determinarán los tipos de mantenimiento, el nivel de servicio y el nivel de capacidad que serán soportados por el proceso de mantenimiento, respetando siempre la estructura establecida por Agil_MANTEMA.

3 Proceso de mantenimiento

Debido a restricciones de espacio, a continuación nos limitamos a hacer una descripción general del proceso de mantenimiento y una descripción específica de una actividad a título de ejemplo, pero en [7] se puede consultar un informe técnico que presenta en detalle Agil_MANTEMA y su proceso de mantenimiento.

A la hora de entender mejor el proceso de mantenimiento de Agil_MANTEMA debe recordarse que está ideado para pequeñas organizaciones en el contexto del proyecto COMPETISOFT, de forma que dicho proceso de mantenimiento forma parte, junto con los *Procesos de desarrollo y Administración de proyectos específicos*, del grupo de procesos de operación del modelo de referencia de procesos propuesto en dicho proyecto iberoamericano [6].

3.1 Descripción

El proceso de mantenimiento tiene como propósito definir una guía explícita para realizar las modificaciones solicitadas en un producto software detallando qué debe realizarse, cuándo, cómo y por quién. Es decir, busca guiar paso a paso el mantenimiento del software para pequeñas organizaciones. En la Fig. 2 se muestra el flujo de trabajo de este proceso en Agil_MANTEMA.

De manera muy resumida, las principales características del proceso de mantenimiento en Agil_MANTEMA son:

- Comienza cuando el “Mantenedor” y el “Cliente” empiezan a trabajar en conjunto, es decir, asignan responsables, criterios y explicación de cómo se va a trabajar. Estas tareas se agrupan en la actividad llamada *Planificación del proceso*.
- Al concluir la actividad anterior se inicia el ciclo que cada “Petición de mantenimiento” recibida tendrá que seguir. La primera actividad a realizar en dicho ciclo es *Atención de la petición*, mediante la cual se formula ó recibe la “Petición de mantenimiento”, que luego

pasa a manos del “Gestor de Peticiones” que se encargará de asignar el tipo y prioridad de la petición. El grupo ordenado de peticiones se llama “Registro de Peticiones”. Desde este registro se selecciona un primer grupo de peticiones llamado “Lista de Espera de Peticiones”. Este grupo puede entrar a dos diferentes tipos de Sprint’s de Mantenimiento (SprintM)¹, uno corto para las peticiones del tipo no planificable y otro más largo para las del tipo planificable. Dentro del SprintM se realizarán una serie de reuniones con el fin de obtener su estado de avance y posibles problemas que puedan ocurrir dentro de su ejecución. Cuando la lista de peticiones seleccionada en un SprintM ha sido resuelta, se finaliza el ciclo con la actividad *Finalizar intervención*, cuya finalidad es la validación y verificación del producto por parte del cliente, su paso a producción, registro de documentos y reuniones de retrospcción.

- Para finalizar el proceso de mantenimiento, cuando ya no se van a recibir más peticiones porque ya se acabó el tiempo del proyecto/servicio, se cuenta con una actividad final llamada *Finalización del servicio*, que sirve para una cesión de actividades por parte del “Mantenedor” de forma que no repercuta negativamente en la organización cliente. En algunas ocasiones, antes de llevar a cabo la finalización es necesario realizar la actividad de *Retirada* con el fin de aplicar el plan de retirada del software.

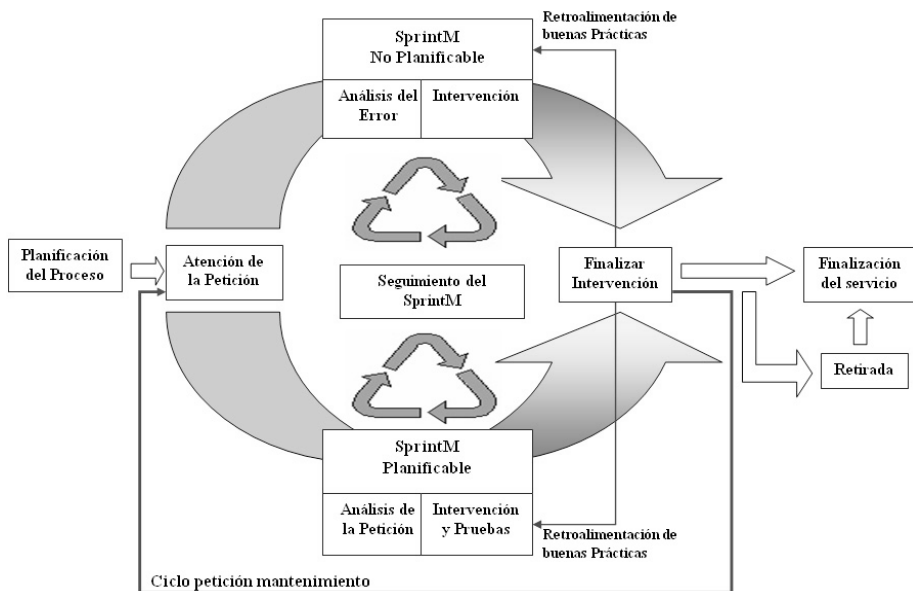


Fig. 2. Flujo de actividades del proceso de mantenimiento en Agil_MANTEMA

Las actividades de *Planificación del proceso*, *Retirada* y *Finalización del servicio* se desarrollan tan solo una vez y no entorpecen la agilidad del proceso. Por otro lado, están los dos tipos de SprintM, cada uno con su conjunto repetitivo de actividades. Siguiendo las directrices de Scrum, existe un mecanismo para incorporar los cambios gracias a que está prevista una realimentación rápida con el cliente, junto con una entrega rápida y periódica de atención a las peticiones de mantenimiento.

Las organizaciones y roles involucrados en el proceso de mantenimiento son:

- Cliente: Es la organización propietaria del software que recibe el servicio de mantenimiento.

¹ SprintM: Ciclo de mantenimiento básico de duración recomendada dependiendo del tipo de mantenimiento (para correctivo urgente de entre uno y siete días, para los otros de entre ocho y quince días) en el que se atiende y resuelve una petición de mantenimiento. (Definición creada para el mantenimiento de software basada en la definición de Sprint de SCRUM).

- Propietario del Producto: Representa a todos los interesados en el producto final. Sus áreas de responsabilidad son: la financiación del proyecto, retorno de la inversión del proyecto y el lanzamiento del proyecto. El propietario del producto por lo general formula peticiones de modificación de tipo perfectivo o adaptativo.
- Usuario: Es quien utiliza el software. Propone las peticiones de modificación correctivas (urgentes o no urgentes) y perfectivas.
- Mantenedor: Es quien realiza la modificación del software.
 - Gestor de Peticiones: Es quien acepta o rechaza las peticiones de modificación y decide el tipo de mantenimiento que corresponde. En caso de ser perfectivo pone al tanto al propietario del producto (cliente) para ver la viabilidad del mantenimiento. En caso de ser cualquiera de los otros tipos, sitúa la petición en la “Lista de Espera de Peticiones”, asignándole una prioridad.
 - Responsable de Mantenimiento: Prepara el proceso y establece las normas y procedimientos necesarios para aplicar la metodología y es el responsable de que se lleven a cabo las prácticas, valores y reglas de Scrum. Además, es miembro del equipo de mantenimiento y trabaja a la par con el resto de miembros, coordina los encuentros permanentes del equipo, y se encarga de eliminar eventuales obstáculos.
 - Equipo de Mantenimiento: Es el grupo de personas que implementa las peticiones de mantenimiento. Tiene autoridad para reorganizarse y definir las acciones necesarias o sugerir remoción de impedimentos.

3.2 Ejemplo de actividad

A manera de ejemplo, a continuación se muestran explícitamente las sub-actividades y tareas de la actividad *SprintM No Planificable*, detallando las entradas, salidas, personal responsable y técnicas de cada tarea. El propósito del *SprintM No Planificable* es brindar atención urgente a las peticiones de modificación que bloquean o interrumpen el funcionamiento del producto software, es decir, este SprintM se ejecuta cuando se asume un mantenimiento correctivo urgente. Ésta actividad se realiza cuando el error² presentado en la petición de modificación paraliza de manera seria el funcionamiento normal del sistema o el de la organización, de forma que la corrección del error debe ser inminente. Se recomienda la ejecución de *SprintM No Planificables* cortos, de entre uno y siete días (dependiendo del tipo de error), con reuniones de gestión todos los días.

Ésta actividad esta compuesta por las dos sub-actividades siguientes (ver también Fig. 3):

- Sub-Actividad SNP1. *Análisis del error*. Consta de una única tarea:
 - Tarea SNP1.1. *Investigar y Analizar Causas*. El “Equipo de mantenimiento” analiza la “Petición de Modificación”, verifica el problema con la colaboración del “Usuario” que realizó la petición y lo reproduce. Además estudia diferentes alternativas para implementar la modificación para la corrección del error. También se construye una lista de los elementos software a corregir (módulos, rutinas, documentos, etc.).
- Sub-Actividad SNP2. *Intervención correctiva urgente*. Ésta actividad consta de dos tareas:
 - Tarea SNP2.1. *Realizar acciones correctivas*. El “Equipo de mantenimiento” ejecuta las acciones necesarias para corregir el problema detectado. Se deben identificar todos los componentes del producto software (rutinas, bases de datos, etc.), afectados por la intervención.
 - Tarea SNP2.2. *Ejecutar pruebas unitarias*. El “Equipo de mantenimiento” debe comprobar el correcto funcionamiento de todos los cambios realizados. Las pruebas realizadas se deben documentar en el documento de “Pruebas Unitarias Realizadas”. Esta tarea sirve para comprobar la correcta operación del módulo al cual se le han llevado a cabo las acciones correctivas.

² Fallo (problema ó defecto) del producto software encontrado en el código ejecutable

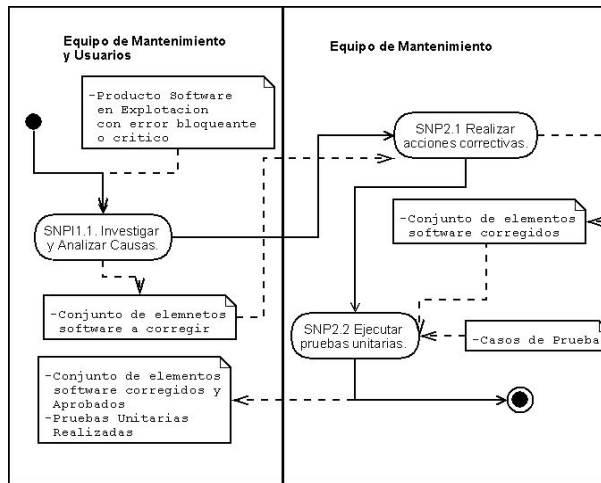


Fig. 3. Diagrama de actividades del *SprintM No Planificable*

En cada una de las tareas incluidas en Agil_MANTEMA se detallan sus correspondientes productos de entrada, productos de salida, técnicas que pueden ser utilizadas para conducir la tarea, roles involucrados, relación con otros procesos y el nivel de servicio que soporta o apoya (ver Tabla 3).

Tabla 3. Ejemplo de descripción de una tarea

	Entradas	Salidas	Técnicas	Roles	Interfaces con otros Procesos	Nivel de Servicio
Tarea SNP1.1 Investigar y Analizar Causas	- Producto Software en explotación con error crítico. - Petición de Modificación	Conjunto de elementos Software a corregir	- Estudio de la documentación - Investigar el Producto Software - Observación y entrevistas	- Equipo de Mantenimiento - Usuario	Soporte al cliente	Básico

4 Caso de estudio

Para el refinamiento y mejora de los resultados del proyecto iberoamericano COMPETISOFT se ha empleado el método de Investigación – Acción (*Action-Research*), dividiendo los participantes del proyecto en dos grupos: el primero constituido por investigadores de distintas universidades y el segundo, denominado grupo crítico de referencia, englobando a profesionales informáticos de las pequeñas organizaciones software e instituciones de estandarización participantes en el proyecto (ver Fig. 4).

En el proyecto COMPETISOFT Agil_MANTEMA versión 1.0 ha sido utilizado como proceso de referencia en dos ciclos de mejora cuyo objetivo fue mejorar el proceso de mantenimiento de software en dos empresas de software de Uruguay (grupo crítico de referencia), realizados durante los meses de octubre de 2007 a abril de 2008. Estos ciclos de mejora, en términos de investigación-acción, fueron llevados adelante por el Grupo de Ingeniería de Software de la Facultad de Ingeniería de la Universidad de la República de Uruguay (investigadores). En los ciclos de mejora se aplicó el marco de modelos elaborado por el proyecto COMPETISOFT, siguiendo como guía los pasos establecidos por el proceso de mejora PmCOMPETISOFT [8] y además utilizaron la propuesta de Agil_MANTEMA (como referencia de buenas prácticas de mantenimiento) para orientar la mejora del proceso de mantenimiento en cada una de las empresas.

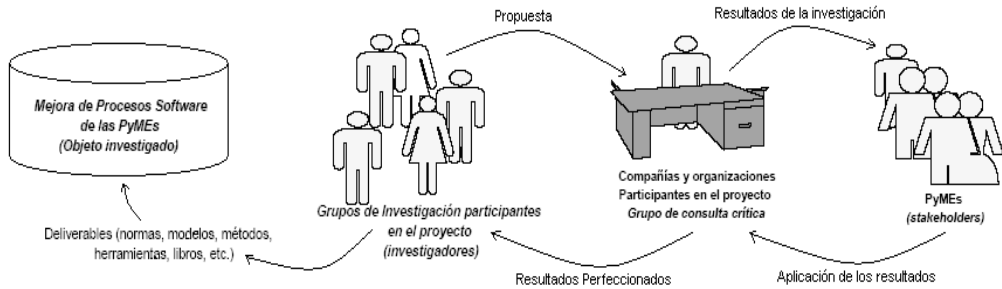


Fig. 4. Método de Investigación-Acción aplicado al proyecto COMPETISOFT

Una de las actividades establecidas por PmCOMPETISOFT es la realización de un *Diagnóstico de procesos* de la situación inicial de la empresa, contrastando los procesos que realiza la empresa contra un modelo de referencia (en este caso Agil_MANTEMA). Este diagnóstico permitió obtener información relevante acerca del estado general del proceso de mantenimiento realizado por cada una de las empresas y tal información fue analizada para generar los resultados del diagnóstico con el objetivo de establecer las oportunidades de mejora de este proceso (las oportunidades de mejora son buenas prácticas que la empresa no realiza en su proceso). Estas buenas prácticas fueron incorporadas en el proceso de mantenimiento de las empresas por medio de las actividades de *Formulación y Ejecución de Mejoras* de PmCOMPETISOFT.

A continuación nosotros nos centramos en presentar del caso de estudio las características de las empresas, las oportunidades de mejora encontradas y llevadas a cabo en los procesos de mantenimiento de las empresas y finalmente las lecciones aprendidas.

4.1 Características de las empresas

Para respetar los requerimientos de confidencialidad de las empresas involucradas, denominamos a dichas empresas como EU1 y EU2. Las principales características de dichas empresas son:

- EU1 tiene 18 años en el mercado y cuenta actualmente con 45 empleados. Tiene un único producto software, desarrollado internamente. Este producto es sumamente adaptable a las características de distintas organizaciones y la empresa ha sido muy exitosa en su comercialización, por lo que al día de hoy cuenta con una muy amplia base instalada. La adaptabilidad del producto ha posibilitado implantarlo en distintos ambientes organizacionales, sin necesidad de modificar los programas. Sin perjuicio de ello, bastantes clientes solicitan modificaciones o agregados especiales al producto. En EU1 se analizó el proceso utilizado para el mantenimiento del único producto de la empresa.
- EU2 tiene 15 años en el mercado y cuenta actualmente con 26 empleados. Ha desarrollado diversos sistemas software, algunos a medida para un cliente en particular, y otros que dieron origen a productos algo más estandarizados que fueron implantados en un número reducido de clientes. En general, los clientes de la empresa son grandes organizaciones y cada instalación tiene un número considerable de usuarios. En EU2 se analizó el proceso utilizado en el proyecto que la empresa consideró que estaba mejor organizado y que funcionaba mejor. Se trata de un sistema desarrollado para una gran organización, utilizado por varias decenas de usuarios y de cuyo correcto funcionamiento depende la calidad de servicio que la organización presta a miles de clientes.

Para ambas empresas se analizaron los roles existentes asignados a tareas de mantenimiento y su correspondencia con los roles definidos por Agil_MANTEMA.

- En EU1, existe el rol de “Analista Funcional” que actúa como representante del cliente y que en parte asume el rol de “Propietario del producto” definido en Agil_MANTEMA, sobre todo en aquellos aspectos del producto que son aplicables a la totalidad de clientes y

usuarios. Por otro lado, existe una persona que cumple a la vez los roles de “Responsable de Mantenimiento” y “Gestor de peticiones”. Se identifica además un “Equipo de Mantenimiento”.

- Por el contrario, en EU2 el “Propietario del Producto” está estrictamente del lado del cliente. Existe también el rol de “Analista Funcional”, que es a su vez “Gestor de Peticiones” y “Responsable de Mantenimiento”.

Respecto a los Tipos de Mantenimiento definidos por Agil_MANTEMA, ambas empresas ofrecen el nivel avanzado que incluye: Correctivo Urgente, Correctivo No Urgente, Perfectivo, Adaptativo y Preventivo.

4.2 Oportunidades de mejora

A continuación se presenta las oportunidades de mejora encontradas al contrastar las actividades previstas por Agil_MANTEMA con las realizadas en las dos empresas. Estas oportunidades de mejora fueron incorporadas a los procesos de mantenimiento de las empresas por medio del ciclo de mejora que cada una de las organizaciones llevó a cabo.

- Definir formalmente el proceso de mantenimiento en ambas organizaciones. Este proceso fue mejorado con prácticas descritas en Agil_MANTEMA ya que en ambas empresas se llevaba a cabo algunas de las actividades establecidas por la metodología propuesta, pero el proceso de mantenimiento no estaba formalmente definido.
- Definición de una estrategia para la gestión de peticiones (requisitos) de modificación para EU1. En EU1, antes de encarar el ciclo de mejora de mantenimiento, se encaró un ciclo de mejora del proceso de desarrollo, como resultado del cual se sistematizó la documentación de los requisitos del sistema y la especificación de los requisitos aplicables a cada requisito o petición. Al encarar la mejora del mantenimiento, se definió la forma de actualizar la especificación de los requisitos aplicables a programas modificados, esencialmente a través de la generación de nuevas versiones de los documentos involucrados. Actualmente se está resolviendo cómo manejar la actualización de los requisitos del sistema, sobre todo para el caso de requisitos que hayan cambiado, ya que generar una nueva versión por cada SprintM resulta en una carga de trabajo muy alta que es difícil que la empresa asuma. Una alternativa que se ha evaluado es mantener la versión original del documento de requisitos y complementarla con unos “delta” correspondientes a las peticiones atendidas.
- Desarrollar un nuevo sistema para la gestión de peticiones para EU2, debido a que las vías de comunicación en esta empresa son diversas (teléfono, correo electrónico, verbal) y no existen garantías de que todas las peticiones sean atendidas. Este sistema se desarrolló durante el ciclo de mejora y actualmente está próximo a entrar en producción. Este sistema tiene como objetivos: (i) transformarse en la única vía de comunicación relacionada con peticiones de mantenimiento y (ii) que un usuario pueda registrar y ver el estado de las peticiones a través de una interfaz Web muy amigable y fácil de usar.
- Introducir un documento para la trazabilidad de las modificaciones realizadas durante una intervención de mantenimiento. El registro de la intervención se realiza en EU1 a medida que avanza el tratamiento de una petición, utilizando el sistema de gestión de peticiones. Como mejora, surgió la conveniencia de incorporar un documento en el que queden registrados los cambios efectivamente realizados, identificando los componentes involucrados. Para el caso de mantenimientos correctivos, esto permite identificar módulos propensos a ser corregidos. En el registro final de mantenimientos correctivos se indica la fase (requerimientos, diseño, construcción) en la que se inyectó el error, con el fin de orientar la mejora del proceso de desarrollo a la fase que sea la mayor fuente de problemas.
- Definición de un documento para la especificación de modificaciones e incorporación de la actividad de *Seguimiento del SprintM* por parte del “Equipo de Mantenimiento” de la empresa EU2 con el fin de evaluar la marcha del Sprint de mantenimiento. En ambas

empresas se realizaba un seguimiento pero no bajo la forma de “Reuniones habituales” al estilo Scrum como propone Agil_MANTEMA. En EU2 se incorporó esta práctica para que todo el equipo esté alineado y aproveche sus propias capacidades para remover los obstáculos que eventualmente encuentren sus integrantes en la realización de sus tareas.

- Incorporación de la *Reunión de retrospectión* propuesta por Agil_MANTEMA por parte de la empresa EU2, ya que ninguna de las dos empresas se realizaba ésta actividad.

4.3 Lecciones aprendidas

Según las dos empresas participantes en el caso de estudio la utilización de Agil_MANTEMA como modelo de referencia para el proceso de mantenimiento constituyó una ayuda muy importante para facilitar la reflexión acerca de su situación y de cara a abordar la mejora de este proceso. Por parte de los investigadores se observó que la propuesta establecida por Agil_MANTEMA resulta adecuada como proceso de referencia para la mejora del proceso de mantenimiento en pequeñas empresas, y además que el enfoque de Scrum es consistente con la visión que el personal de mantenimiento de estas empresas tiene del proceso de mantenimiento.

De estas dos experiencias en entornos reales se puede resaltar que la utilización de Agil_MANTEMA fue enriquecedora, tanto para las empresas como para el grupo de mejora encargado de la aplicación, y el grupo de investigadores creadores de la metodología.

Algunas lecciones aprendidas relevantes extraídas de estos dos casos de uso son:

- Entre las cuestiones que más preocupaban al Responsable de Mantenimiento de EU1 se encontraba el poder determinar y gestionar los compromisos asumidos por el equipo de mantenimiento para atender las demandas de los diversos clientes y avisar con tiempo en caso de que resultara imposible cumplir los plazos acordados. En Agil_MANTEMA no aparecen actividades que permitan atender estas preocupaciones porque considera que el mantenimiento en una pequeña organización debe tratarse como un proyecto más. Así, en el modelo de referencia creado en el proyecto COMPETISOFT el proceso de *Administración de proyectos específicos* es el encargado de la gestión del proyecto de mantenimiento, y debe utilizarse junto con Agil_MANTEMA cuando se lleve a cabo acciones de mantenimiento de software.
- Al analizar el panorama de las peticiones en EU1, el Responsable de Mantenimiento detectó algunas que estaban pendientes desde el año 2005, lo que mostró la conveniencia de introducir nuevos elementos a considerar relacionados con el análisis de las peticiones de Agil_MANTEMA. Entre otros aspectos, corresponde analizar la carga de trabajo comprometida, su evolución, la concentración de peticiones en determinados componentes (especialmente en el caso de mantenimiento correctivo), las prioridades, la vigencia de las peticiones reportadas hace mucho tiempo y las peticiones que pueden estar relacionadas. Otra vez, estos aspectos están más relacionados con la gestión de proyectos/servicios que con el propio mantenimiento.
- Los Responsables de Mantenimiento en ambas empresas consideraron que el enfoque general de Scrum involucrado en Agil_MANTEMA y el énfasis que pone en planificar a corto plazo en función de las prioridades de las *Peticiones de mantenimiento* existentes en un momento específico de tiempo, permite combinar de manera flexible objetivos conflictivos como son la búsqueda de eficiencia en la ejecución y la adecuada atención de los cambios en las prioridades de tales peticiones.
- En ambas empresas consideran que el agrupamiento de peticiones para una planificación a corto plazo, como el presentado en el *SprintM Planificable*, ofrece beneficios como: (i) permitir evaluar la compatibilidad y consistencia de los cambios asociados a un conjunto de peticiones y (ii) permitir mejorar la eficiencia en la ejecución ya que es posible realizar una única instancia de pruebas de regresión para el grupo de peticiones.
- A partir del análisis de los procesos de mantenimiento en ambas empresas, queda claro que los arreglos de emergencia debieran encarrilarse a posteriori en un *SprintM Planificable*, para

garantizar que las intervenciones realizadas mantienen el nivel de calidad requerido, completar las pruebas eventualmente no realizadas y asegurar que el problema que originó la petición urgente queda solucionado en las versiones posteriores del software.

- La gestión del mantenimiento involucra mucha información proveniente de diversas fuentes, por lo que contar con un Sistema de Gestión de Peticiones que permita manejar y gestionar la documentación de las intervenciones resulta de gran ayuda para la implantación de cambios durante el proceso de mantenimiento.

Adicionalmente al aprendizaje y reflexión por parte de las empresas, el grupo encargado de la creación de la metodología obtuvo una primera evaluación de su utilización en un entorno real. Con la realimentación brindada por las personas de las empresas que utilizaron la metodología se recopiló información valiosa para proceder a realizar algunas mejoras en la misma. Tales mejoras estarán enfocadas a:

- Proporcionar tareas de soporte a la comunicación entre el equipo de mantenimiento y el usuario ya que es un punto neurálgico en el proceso de mantenimiento
- Incorporar en un SprintM tareas de verificación y validación de la consistencia del conjunto de “Peticiones de mantenimiento” involucradas, como un todo.
- Sustituir la tarea *Ejecutar paralelamente el software* de un *SprintM Planificable* por “Pruebas de Regresión”, que resulta más general y adecuada.
- Indicar explícitamente que una petición urgente que da inicio a un *SprintM No Planificable* puede recibir atención unitaria, es decir que puede no ser agrupada con ninguna otra petición.
- Incorporar en el registro de la actividad *Finalizar Intervención* del mantenimiento correctivo, la indicación de la fase (análisis, diseño, construcción, prueba) en la que se inyectó el error.
- Extraer la tarea *Pasar a producción* de la actividad *Finalizar Intervención*, ya que es frecuente que la puesta en producción de un producto software modificado es una actividad independiente y distinta a la de finalización de la intervención.
- Crear plantillas predefinidas aplicables a las diferentes actividades establecidas por la metodología para facilitar la implantación de Agil_MANTEMA.

Actualmente las lecciones aprendidas y las sugerencias para la mejora de la metodología están siendo analizadas con el fin de incorporarlas en una nueva versión de Agil_MANTEMA.

5 Conclusiones y trabajos futuros

En este artículo se ha presentado Agil_MANTEMA que propone una metodología de mantenimiento de software enfocada a pequeñas organizaciones. Esta metodología establece los elementos necesarios para guiar paso a paso el mantenimiento de software enfocado a pequeñas organizaciones. Además esta metodología ha sido utilizada por dos pequeñas empresas que llevaron a cabo un ciclo de mejora centrado en el proceso de mantenimiento. De la utilización inicial de Agil_MANTEMA en las dos pequeñas empresas se observa que la metodología de mantenimiento propuesta es adecuada a este tipo de empresas, ya que ambas empresas han definido formalmente su proceso de mantenimiento a partir de Agil_MANTEMA con el objetivo de mejorar su rutina diaria relacionada con el mantenimiento de software. Entre los aspectos más relevantes a destacar en cada uno de las empresas se tiene:

- En EU1, al diagnosticar su proceso de mantenimiento con Agil_MANTEMA encontraron que era posible reconocer cómo encajaban gran parte de las prácticas y actividades llevadas a cabo en la empresa, en algún lugar de Agil_MANTEMA. Sin embargo, quedó claro que el proceso de mantenimiento seguido por la empresa adolecía de diferentes buenas prácticas que sugiere la metodología de mantenimiento (las cuales no se realizaban por la empresa). Para mejorar el proceso de mantenimiento se trabajó en 4 aspectos interrelacionados: (i) delimitar roles y responsabilidades, (ii) mejorar mecanismos de comunicación y

documentación, (iii) sistematizar la documentación de las intervenciones y (iv) ajustar el sistema para la gestión de las peticiones.

- Por su parte en EU2, al diagnosticar el proceso de mantenimiento del proyecto analizado, con Agil_MANTEMA, encontraron que ya realizaba agrupamientos de peticiones de acuerdo a un SprintM. También, como en EU1 detectaron que sus mayores carencias estaban relacionadas con el manejo y gestión de las solicitudes de mantenimiento y su intervención, por lo que sus actividades de mejora estuvieron encaminadas a fortalecer estos aspectos. Así pues, esta empresa que no disponía de un Sistema de Gestión de Peticiones, optó por crear uno para que clientes y usuarios pudieran registrar y consultar el estado de las peticiones, con el objetivo de canalizar toda la comunicación con el Equipo de Mantenimiento. Adicional a este aspecto también incorporaron prácticas ágiles de seguimiento a las actividades del proceso de mantenimiento.

Como trabajo futuro se incorporarán las sugerencias de mejora obtenidas de este caso de estudio en una nueva versión de Agil_MANTEMA. Además se aplicará esta metodología en otras pequeñas empresas que participan en el proyecto COMPETISOFT. El objetivo es que a partir de un conjunto representativo de casos de uso se obtenga nueva realimentación para proceder a la evaluación, refinamiento, mejora y validación de la metodología de mantenimiento propuesta.

Agradecimientos

Este trabajo ha sido financiado por los proyectos COMPETISOFT (506AC0287) del programa CYTED; MECENAS (PBI06-0024) e INGENIO (PAC08-0154-9262) de la Junta de Comunidades de Castilla-La Mancha, Consejería de Educación y Ciencia; y ESFINGE (TIN2006-15175-C05-05) del Ministerio de Ciencia y Tecnología de España.

Referencias

- [1] Abrahamsson, P., O. Salo, J. Rankainen, and J. Warsta, *Agil software development methods: review and analysis*. 2002, Finland, VTT Publications 478.
- [2] Batista, J. and A. Figueiredo, *SPI in a very small team: a case with CMM*. Software Process: Improvement and Practice, 2000. Vol. 5(4) December pp. 243-250.
- [3] ISO. *ISO/IEC 15504-2:2003/Cor.1:2004(E). Information technology - Process assessment - Part 2: Performing an assessment*. International Organization for Standardization. Geneva. 2004. www.iso.org
- [4] MAP. *Métrica Versión 3. Metodología de Planificación, Desarrollo y Mantenimiento de sistemas de información*. 2007. <http://www.csi.map.es/csi/metrica3/index.html>
- [5] Mayer&Bunge. *Panorama de la Industria del Software en Latinoamérica*. Mayer & Bunge Informática LTDA. Brasil. 2004. www.mbi.com.br/200409_panorama_industria_software_america_latina.pdf
- [6] Oktaba, H., F. Garcia, M. Piattini, F. Pino, C. Alquicira, and F. Ruiz, *Software Process Improvement: The COMPETISOFT Project*. IEEE Computer, 2007. Vol. 40(10) October pp. 21-28.
- [7] Pino, F., F. Ruiz, and S. Salas. *Agil_Mantema Versión 1.0. COMPETISOFT IT_23*. CYTED. Ciudad Real, España. 2008. http://alarcos.inf-cr.uclm.es/competisoft/Agil_MANTEMA_Ver1.pdf
- [8] Pino, F., J. Vidal, F. Garcia, and M. Piattini. *Modelo para la Implementación de Mejora de Procesos en Pequeñas Organizaciones Software*. 2007. XII Jornadas de Ingeniería del Software y Bases de Datos, JISBD'2007. pp. 326-335.
- [9] Polo, M., M. Piattini, F. Ruiz, and C. Calero. *MANTEMA: A Software Maintenance Methodology Based on the ISO/IEC 12207 Standard*. 1999. Proceedings of the 4th IEEE International Symposium and Forum on Software Engineering Standards. Curitiba, Brazil. pp. 76-81.
- [10] Ruiz, F., M. Piattini, M. Polo, and C. Calero. *Maintenance types in the MANTEMA Methodology*. 1999. Int. Conf. on Enterprise Information System. Setubal, Portugal. pp. 27-30.
- [11] Takeuchi, H. and I. Nonaka, *The New New Product Development Game*. Harvard Business Review, 1986 Jan.

