7th IEEE INTERNATIONAL CONFERENCE ON GLOBAL SOFTWARE ENGINEERING

PORTO ALEGRE, **BRAZIL**
PUCRS

27-30 August
2012

www.icgse.org

CONFERENCE INFORMATION

PAPERS BY SESSION

PAPERS BY AUTHOR

GETTING STARTED

TRADEMARKS

SEARCH

Published by

CPS
Conference Publishing Services

Conference Information

2012 IEEE Seventh International Conference on Global Software Engineering

- □ Message from the General Chair
- □ Message from the Program Chairs
- □ Organizing Committee
- □ Program Committee
- □ Reviewers
- □ Title Page (Book version)
- □ Copyright Page (Book version)
- □ Table of Contents (Book version)
- □ Author Index (Book version)
- □ Publisher's Information (Book version)

PREVIOUS HOME NEXT

Sessions

- □ Best Paper Candidates
- □ From the Trenches
- □ Tools and Methods
- □ Experience Based
- □ Systematic Literature Reviews: What are We Learning?
- □ Academic Short Papers
- □ Academic New Ideas
- □ Position Papers
- □ Industry Track

PREVIOUS HOME NEXT

Papers by Session

Best Paper Candidates

- □ From Offshore Outsourcing to Offshore Insourcing: Three Stories
  Nils Brede Moe, Darja Šmite, and Geir Kjetil Hanssen
- □ Agile vs. Structured Distributed Software Development: A Case Study
  H.-Christian Estler, Martin Nordio, Carlo A. Furia, Bertrand Meyer, and Johannes Schneider
- □ Activity Theory Applied to Global Software Engineering: Theoretical Foundations and Implications for Tool Builders
  Paolo Tell and Muhammad Ali Babar

PREVIOUS
HOME
NEXT

Papers by Session

From the Trenches

□ From RUP to Scrum in Global Software Development: A Case Study
Ramon Noordeloos, Christina Manteli, and Hans Van Vliet

□ An Exploratory Study of Software Evolution and Quality: Before, during and after a Transfer
Ronald Jabangwe and Darja Šmite

□ Configuration Management Stories from the Distributed Software Development Trenches
Lars Bendix, Jan Magnusson, and Christian Pendleton

PREVIOUS

HOME

NEXT

Papers by Session

Tools and Methods

❑ FTSProc: A Process to Alleviate the Challenges of Projects that Use the Follow-the-Sun Strategy
Estevão Ricardo Hess and Jorge Luis Nicolas Audy

❑ An Industrial Evaluation of Technological Support for Overhearing Conversations in Global Software Engineering
Kevin Dullemond and Ben van Gameren

❑ 2DAM-WAVE: An Evaluation Method for the WAVE Capability Model
Rafael A. Glanzner and Jorge Luis Nicolas Audy

PREVIOUS HOME NEXT

Papers by Session

Experience Based

- ❑ Supporting Acceptance Testing in Distributed Software Projects with Integrated Feedback Systems: Experiences and Requirements
  Olga Liskin, Christoph Herrmann, Eric Knauss, Thomas Kurpick, Bernhard Rumpe, and Kurt Schneider
- ❑ A Tool Framework for Deriving the Application Architecture for Global Software Development Projects
  Bugra M. Yildiz, Bedir Tekinerdogan, and Semih Cetin
- ❑ Distributed Developers and the Non-use of Web 2.0 Technologies: A Proclivity Model
  Ban Al-Ani, Yi Wang, Sabrina Marczak, Erik Trainer, and David Redmiles

PREVIOUS HOME NEXT

Papers by Session

Systematic Literature Reviews: What are We Learning?

- □ On the Nature of GSE Organizational Social Structures: An Empirical Study
  Damian A. Tamburri, Patricia Lago, Hans Van Vliet, and Elisabetta di Nitto
- □ Adopting a Social Network Perspective in Global Software Development
  Christina Manteli, Hans van Vliet, and Bart van den Hooff
- □ Systematic Literature Reviews in Distributed Software Development: A Tertiary Study
  Anna Beatriz Marques, Rosiane Rodrigues, and Tayana Conte

PREVIOUS HOME NEXT

Papers by Session

Academic Short Papers

- □ The Many Facets of Distance and Space: The Mobility of Actors in Globally Distributed Project Teams
  Tony Clear, Waqar Hussain, and Stephen G. Macdonell

- □ Near-Synchronicity and Distance: Instant Messaging as a Medium for Global Software Engineering
  Tuomas Jaanu, Maria Paasivaara, and Casper Lassenius

- □ Problems? We All Know We Have Them. Do We Have Solutions Too? A Literature Review on Problems and Their Solutions in Global Software Development
  Vanessa Gomes and Sabrina Marczak

PREVIOUS HOME NEXT

Papers by Session

Academic New Ideas

❑ A Qualitative User Study on Preemptive Conflict Detection
Lile Hattori, Michele Lanza, and Marco D'Ambros

❑ Mapping Global Software Development Practices for Follow-the-Sun Process
Josiane Kroll and Jorge Luis Nicolas Audy

❑ Enterprise Architecture Cybernetics for Complex Global Software Development: Reducing the Complexity of Global Software Development Using Extended Axiomatic Design Theory
Hadi Kandjani, Peter Bernus, and Lian Wen

❑ Experiences in Scaling the Product Owner Role in Large-Scale Globally Distributed Scrum
Maria Paasivaara, Ville T. Heikkilä, and Casper Lassenius

PREVIOUS

HOME

NEXT

Papers by Session

Position Papers

❑ Setting Up a Stochastic Model for Teams Working in a Follow-the-Sun Environment
Alan R. Santos, Afonso Sales, and Paulo Fernandes

❑ Prerequisites for Teaching of Distributed Software Development in Graduate Courses
Mariana Orlandini Ferreira, Alexandre L'Erario, and José Augusto Fabri

❑ Task Allocation Model in Globally Distributed Software Projects Using Genetic Algorithms
Jude Fernandez and M. Basavaraju

❑ A Systematic Tertiary Study of Communication in Distributed Software Development Projects
Alinne C.C. dos Santos, Ivaldir H. de Farias Junior, Hermano P. de Moura, and Sabrina Marczak

❑ Social Awareness for Global Software Teams
Fabio Calefato and Filippo Lanubile

❑ A Distributed Software Development Environment Dynamics Model
Alexandre L'Erario, José Augusto Fabri, André Domingues, Alessandro Duarte, Rodrigo Palácios, and Marcelo Schneck de Paula Pessôa

PREVIOUS HOME NEXT

Papers by Session

Industry Track

□ Managing Global R&D Projects: Practical Experience in Building Project Management Competency
Rakesh Singh and Kurt Hofmann

□ An Industrial Experience on the Application of Distributed Testing in an Agile Software Development Environment
Eliane Collins, Gisele Macedo, Nayane Maia, and Arilo Dias-Neto

□ Subcontracting for Scrum Teams: Experiences and Guidelines from a Large Development Organization
Miika Penttinen and Tommi Mikkonen

□ Challenges in Distributed Scrum
Gopal K. Ghosh

□ Freelancers: A Global Software Engineering Approach for Small Projects
Abhinandan Bhadauria

□ Process Harmonization across Remote Sites
Manoj Apte and Kurt Hofmann

PREVIOUS

HOME

NEXT

Papers by Session

- ❑ 4 C: An Approach for Effective People Management in an Offshore Software Development Center
  Gerd Höfner and V.S. Mani
- ❑ An Approach to Effectively Transfer Knowledge and Accelerate the Movement of Software Services Offshore
  Sandeep Kumar, Ajit Krishna, V.S. Mani, B.K. Kulkarni, and Carlos Arglebe
- ❑ Introducing Global Software Development in Turkey: Why and How?
  Bedir Tekinerdogan and Semih Cetin

PREVIOUS HOME NEXT

Papers by Author

A

- □ Al-Ani, Ban
- □ Apte, Manoj
- □ Arglebe, Carlos
- □ Audy, Jorge Luis Nicolas

B

- □ Babar, Muhammad Ali
- □ Basavaraju, M.
- □ Bendix, Lars
- □ Bernus, Peter
- □ Bhadauria, Abhinandan

C

- □ Calefato, Fabio
- □ Cetin, Semih
- □ Clear, Tony
- □ Collins, Eliane
- □ Conte, Tayana

D

- □ D'Ambros, Marco
- □ Dias-Neto, Arilo
- □ Domingues, André
- □ Duarte, Alessandro
- □ Dullemond, Kevin

E

- □ Estler, H.-Christian

PREVIOUS

HOME

NEXT

Papers by Author

F

- ❑ Fabri, José Augusto
- ❑ Fernandes, Paulo
- ❑ Fernandez, Jude
- ❑ Ferreira, Mariana Orlandini
- ❑ Furia, Carlo A.

G

- ❑ Gameren, Ben van
- ❑ Ghosh, Gopal K.
- ❑ Glanzner, Rafael A.
- ❑ Gomes, Vanessa

H

- ❑ Hanssen, Geir Kjetil
- ❑ Hattori, Lile
- ❑ Heikkilä, Ville T.
- ❑ Herrmann, Christoph
- ❑ Hess, Estevão Ricardo
- ❑ Hofmann, Kurt
- ❑ Höfner, Gerd
- ❑ Hooff, Bart van den
- ❑ Hussain, Waqar

J

- ❑ Jaanu, Tuomas
- ❑ Jabangwe, Ronald
- ❑ Junior, Ivaldir H. de Farias

PREVIOUS HOME NEXT

Papers by Author

K

- □ Kandjani, Hadi
- □ Knauss, Eric
- □ Krishna, Ajit
- □ Kroll, Josiane
- □ Kulkarni, B.K.
- □ Kumar, Sandeep
- □ Kurpick, Thomas

L

- □ Lago, Patricia
- □ Lanubile, Filippo
- □ Lanza, Michele
- □ Lassenius, Casper
- □ L'Erario, Alexandre
- □ Liskin, Olga

M

- □ Macdonell, Stephen G.
- □ Macedo, Gisele
- □ Magnusson, Jan
- □ Maia, Nayane
- □ Mani, V.S.
- □ Manteli, Christina
- □ Marczak, Sabrina
- □ Marques, Anna Beatriz
- □ Meyer, Bertrand
- □ Mikkonen, Tommi
- □ Moe, Nils Brede
- □ Moura, Hermano P. de

PREVIOUS HOME NEXT

Papers by Author

N

- ❑ Nitto, Elisabetta di
- ❑ Noordeloos, Ramon
- ❑ Nordio, Martin

P

- ❑ Paasivaara, Maria
- ❑ Palácios, Rodrigo
- ❑ Pendleton, Christian
- ❑ Penttinen, Miika
- ❑ Pessôa, Marcelo Schneck de Paula

R

- ❑ Redmiles, David
- ❑ Rodrigues, Rosiane
- ❑ Rumpe, Bernhard

S

- ❑ Sales, Afonso
- ❑ Santos, Alan R.
- ❑ Santos, Alinne C.C. dos
- ❑ Schneider, Johannes
- ❑ Schneider, Kurt
- ❑ Singh, Rakesh

Š

- ❑ Šmite, Darja

T

- ❑ Tamburri, Damian A.

PREVIOUS

HOME

NEXT

Papers by Author

□ Tekinerdogan, Bedir

□ Tell, Paolo

□ Trainer, Erik

V

□ Vliet, Hans van

□ Vliet, Hans Van

W

□ Wang, Yi

□ Wen, Lian

Y

□ Yildiz, Bugra M.

PREVIOUS

HOME

NEXT

Papers by Author

Al-Ani, Ban

❑ Distributed Developers and the Non-use of Web 2.0 Technologies: A Proclivity Model

Apte, Manoj

❑ Process Harmonization across Remote Sites

Arglebe, Carlos

❑ An Approach to Effectively Transfer Knowledge and Accelerate the Movement of Software Services Offshore

Audy, Jorge Luis Nicolas

❑ FTSProc: A Process to Alleviate the Challenges of Projects that Use the Follow-the-Sun Strategy

❑ 2DAM-WAVE: An Evaluation Method for the WAVE Capability Model

PREVIOUS HOME NEXT

Papers by Author

- ❑ Mapping Global Software Development Practices for Follow-the-Sun Process

Babar, Muhammad Ali

- ❑ Activity Theory Applied to Global Software Engineering: Theoretical Foundations and Implications for Tool Builders

Basavaraju, M.

- ❑ Task Allocation Model in Globally Distributed Software Projects Using Genetic Algorithms

Bendix, Lars

- ❑ Configuration Management Stories from the Distributed Software Development Trenches

PREVIOUS HOME NEXT

Papers by Author

Bernus, Peter

- ❑ Enterprise Architecture Cybernetics for Complex Global Software Development: Reducing the Complexity of Global Software Development Using Extended Axiomatic Design Theory

Bhadauria, Abhinandan

- ❑ Freelancers: A Global Software Engineering Approach for Small Projects

Calefato, Fabio

- ❑ Social Awareness for Global Software Teams

Cetin, Semih

- ❑ A Tool Framework for Deriving the Application Architecture for Global Software Development Projects
- ❑ Introducing Global Software Development in Turkey: Why and How?

PREVIOUS HOME NEXT

Papers by Author

Clear, Tony

❑ The Many Facets of Distance and Space: The Mobility of Actors in Globally Distributed Project Teams

Collins, Eliane

❑ An Industrial Experience on the Application of Distributed Testing in an Agile Software Development Environment

Conte, Tayana

❑ Systematic Literature Reviews in Distributed Software Development: A Tertiary Study

D'Ambros, Marco

❑ A Qualitative User Study on Preemptive Conflict Detection

PREVIOUS HOME NEXT

Papers by Author

Dias-Neto, Arilo

- ❑ An Industrial Experience on the Application of Distributed Testing in an Agile Software Development Environment

Domingues, André

- ❑ A Distributed Software Development Environment Dynamics Model

Duarte, Alessandro

- ❑ A Distributed Software Development Environment Dynamics Model

Dullemond, Kevin

- ❑ An Industrial Evaluation of Technological Support for Overhearing Conversations in Global Software Engineering

Estler, H.-Christian

- ❑ Agile vs. Structured Distributed Software Development: A Case Study

PREVIOUS HOME NEXT

Papers by Author

Fabri, José Augusto

- ❑ Prerequisites for Teaching of Distributed Software Development in Graduate Courses
- ❑ A Distributed Software Development Environment Dynamics Model

Fernandes, Paulo

- ❑ Setting Up a Stochastic Model for Teams Working in a Follow-the-Sun Environment

Fernandez, Jude

- ❑ Task Allocation Model in Globally Distributed Software Projects Using Genetic Algorithms

Ferreira, Mariana Orlandini

- ❑ Prerequisites for Teaching of Distributed Software Development in Graduate Courses

PREVIOUS HOME NEXT

Papers by Author

Furia, Carlo A.

□ Agile vs. Structured Distributed Software Development: A Case Study

Gameren, Ben van

□ An Industrial Evaluation of Technological Support for Overhearing Conversations in Global Software Engineering

Ghosh, Gopal K.

□ Challenges in Distributed Scrum

Glanzner, Rafael A.

□ 2DAM-WAVE: An Evaluation Method for the WAVE Capability Model

PREVIOUS
HOME
NEXT

Papers by Author

Gomes, Vanessa

❑ Problems? We All Know We Have Them. Do We Have Solutions Too? A Literature Review on Problems and Their Solutions in Global Software Development

Hanssen, Geir Kjetil

❑ From Offshore Outsourcing to Offshore Insourcing: Three Stories

Hattori, Lile

❑ A Qualitative User Study on Preemptive Conflict Detection

Heikkilä, Ville T.

❑ Experiences in Scaling the Product Owner Role in Large-Scale Globally Distributed Scrum

PREVIOUS

HOME

NEXT

Papers by Author

Herrmann, Christoph

- □ Supporting Acceptance Testing in Distributed Software Projects with Integrated Feedback Systems: Experiences and Requirements

Hess, Estevão Ricardo

- □ FTSProc: A Process to Alleviate the Challenges of Projects that Use the Follow-the-Sun Strategy

Hofmann, Kurt

- □ Managing Global R&D Projects: Practical Experience in Building Project Management Competency
- □ Process Harmonization across Remote Sites

Höfner, Gerd

- □ 4 C: An Approach for Effective People Management in an Offshore Software Development Center

PREVIOUS HOME NEXT

Papers by Author

Hooff, Bart van den

- ❑ Adopting a Social Network Perspective in Global Software Development

Hussain, Waqar

- ❑ The Many Facets of Distance and Space: The Mobility of Actors in Globally Distributed Project Teams

Jaanu, Tuomas

- ❑ Near-Synchronicity and Distance: Instant Messaging as a Medium for Global Software Engineering

Jabangwe, Ronald

- ❑ An Exploratory Study of Software Evolution and Quality: Before, during and after a Transfer

PREVIOUS HOME NEXT

Papers by Author

Junior, Ivaldir H. de Farias

❑ A Systematic Tertiary Study of Communication in Distributed Software Development Projects

Kandjani, Hadi

❑ Enterprise Architecture Cybernetics for Complex Global Software Development: Reducing the Complexity of Global Software Development Using Extended Axiomatic Design Theory

Knauss, Eric

❑ Supporting Acceptance Testing in Distributed Software Projects with Integrated Feedback Systems: Experiences and Requirements

Krishna, Ajit

❑ An Approach to Effectively Transfer Knowledge and Accelerate the Movement of Software Services Offshore

PREVIOUS HOME NEXT

Papers by Author

Kroll, Josiane

- ❑ Mapping Global Software Development Practices for Follow-the-Sun Process

Kulkarni, B.K.

- ❑ An Approach to Effectively Transfer Knowledge and Accelerate the Movement of Software Services Offshore

Kumar, Sandeep

- ❑ An Approach to Effectively Transfer Knowledge and Accelerate the Movement of Software Services Offshore

Kurpick, Thomas

- ❑ Supporting Acceptance Testing in Distributed Software Projects with Integrated Feedback Systems: Experiences and Requirements

PREVIOUS HOME NEXT

Papers by Author

Lago, Patricia

□ On the Nature of GSE Organizational Social Structures: An Empirical Study

Lanubile, Filippo

□ Social Awareness for Global Software Teams

Lanza, Michele

□ A Qualitative User Study on Preemptive Conflict Detection

Lassenius, Casper

□ Near-Synchronicity and Distance: Instant Messaging as a Medium for Global Software Engineering

□ Experiences in Scaling the Product Owner Role in Large-Scale Globally Distributed Scrum

PREVIOUS HOME NEXT

Papers by Author

L'Erario, Alexandre

- ❑ Prerequisites for Teaching of Distributed Software Development in Graduate Courses
- ❑ A Distributed Software Development Environment Dynamics Model

Liskin, Olga

- ❑ Supporting Acceptance Testing in Distributed Software Projects with Integrated Feedback Systems: Experiences and Requirements

Macdonell, Stephen G.

- ❑ The Many Facets of Distance and Space: The Mobility of Actors in Globally Distributed Project Teams

Macedo, Gisele

- ❑ An Industrial Experience on the Application of Distributed Testing in an Agile Software Development Environment

PREVIOUS HOME NEXT

Papers by Author

Magnusson, Jan

- ❑ Configuration Management Stories from the Distributed Software Development Trenches

Maia, Nayane

- ❑ An Industrial Experience on the Application of Distributed Testing in an Agile Software Development Environment

Mani, V.S.

- ❑ 4 C: An Approach for Effective People Management in an Offshore Software Development Center
- ❑ An Approach to Effectively Transfer Knowledge and Accelerate the Movement of Software Services Offshore

Manteli, Christina

- ❑ From RUP to Scrum in Global Software Development: A Case Study

PREVIOUS
HOME
NEXT

Papers by Author

- Adopting a Social Network Perspective in Global Software Development

Marczak, Sabrina

- Distributed Developers and the Non-use of Web 2.0 Technologies: A Proclivity Model
- Problems? We All Know We Have Them. Do We Have Solutions Too? A Literature Review on Problems and Their Solutions in Global Software Development
- A Systematic Tertiary Study of Communication in Distributed Software Development Projects

Marques, Anna Beatriz

- Systematic Literature Reviews in Distributed Software Development: A Tertiary Study

PREVIOUS
HOME
NEXT

Papers by Author

Meyer, Bertrand

- ❑ Agile vs. Structured Distributed Software Development: A Case Study

Mikkonen, Tommi

- ❑ Subcontracting for Scrum Teams: Experiences and Guidelines from a Large Development Organization

Moe, Nils Brede

- ❑ From Offshore Outsourcing to Offshore Insourcing: Three Stories

Moura, Hermano P. de

- ❑ A Systematic Tertiary Study of Communication in Distributed Software Development Projects

PREVIOUS HOME NEXT

Papers by Author

Nitto, Elisabetta di

- ❑ On the Nature of GSE Organizational Social Structures: An Empirical Study

Noordeloos, Ramon

- ❑ From RUP to Scrum in Global Software Development: A Case Study

Nordio, Martin

- ❑ Agile vs. Structured Distributed Software Development: A Case Study

Paasivaara, Maria

- ❑ Near-Synchronicity and Distance: Instant Messaging as a Medium for Global Software Engineering
- ❑ Experiences in Scaling the Product Owner Role in Large-Scale Globally Distributed Scrum

PREVIOUS

HOME

NEXT

Papers by Author

Palácios, Rodrigo

□ A Distributed Software Development Environment Dynamics Model

Pendleton, Christian

□ Configuration Management Stories from the Distributed Software Development Trenches

Penttinen, Miika

□ Subcontracting for Scrum Teams: Experiences and Guidelines from a Large Development Organization

Pessôa, Marcelo Schneck de Paula

□ A Distributed Software Development Environment Dynamics Model

PREVIOUS HOME NEXT

Papers by Author

Redmiles, David

□ Distributed Developers and the Non-use of Web 2.0 Technologies: A Proclivity Model

Rodrigues, Rosiane

□ Systematic Literature Reviews in Distributed Software Development: A Tertiary Study

Rumpe, Bernhard

□ Supporting Acceptance Testing in Distributed Software Projects with Integrated Feedback Systems: Experiences and Requirements

Sales, Afonso

□ Setting Up a Stochastic Model for Teams Working in a Follow-the-Sun Environment

PREVIOUS HOME NEXT

Papers by Author

Santos, Alan R.

- ❑ Setting Up a Stochastic Model for Teams Working in a Follow-the-Sun Environment

Santos, Alinne C.C. dos

- ❑ A Systematic Tertiary Study of Communication in Distributed Software Development Projects

Schneider, Johannes

- ❑ Agile vs. Structured Distributed Software Development: A Case Study

Schneider, Kurt

- ❑ Supporting Acceptance Testing in Distributed Software Projects with Integrated Feedback Systems: Experiences and Requirements

PREVIOUS
HOME
NEXT

Papers by Author

Singh, Rakesh

- ❑ Managing Global R&D Projects: Practical Experience in Building Project Management Competency

Šmite, Darja

- ❑ From Offshore Outsourcing to Offshore Insourcing: Three Stories
- ❑ An Exploratory Study of Software Evolution and Quality: Before, during and after a Transfer

Tamburri, Damian A.

- ❑ On the Nature of GSE Organizational Social Structures: An Empirical Study

Tekinerdogan, Bedir

- ❑ A Tool Framework for Deriving the Application Architecture for Global Software Development Projects

PREVIOUS HOME NEXT

Papers by Author

- □ Introducing Global Software Development in Turkey: Why and How?

Tell, Paolo

- □ Activity Theory Applied to Global Software Engineering: Theoretical Foundations and Implications for Tool Builders

Trainer, Erik

- □ Distributed Developers and the Non-use of Web 2.0 Technologies: A Proclivity Model

Vliet, Hans van

- □ Adopting a Social Network Perspective in Global Software Development

Vliet, Hans Van

- □ From RUP to Scrum in Global Software Development: A Case Study

PREVIOUS

HOME

NEXT

Papers by Author

- ❑ On the Nature of GSE Organizational Social Structures: An Empirical Study

Wang, Yi

- ❑ Distributed Developers and the Non-use of Web 2.0 Technologies: A Proclivity Model

Wen, Lian

- ❑ Enterprise Architecture Cybernetics for Complex Global Software Development: Reducing the Complexity of Global Software Development Using Extended Axiomatic Design Theory

Yildiz, Bugra M.

- ❑ A Tool Framework for Deriving the Application Architecture for Global Software Development Projects

PREVIOUS HOME NEXT

Papers by Author

A B C D E F G

H I J K L M N

O P Q R S T U

V W X Y Z

PREVIOUS

HOME

2012 IEEE Seventh International Conference on Global Software Engineering Workshops

DPMTool: A Tool for Decisions Management in Distributed Software Projects

Pedro José Garrido, Aurora Vizcaíno, Juan Andrada, Miguel J. Monasor and Mario Piattini
Alarcos Research Group,
Institute of Information Technologies & Systems and Technologies Department, UCLM-INDRA Research and Development Institute,
Escuela Superior de Informática,
University of Castilla-La Mancha
Ciudad Real, Spain
{ Pedro.Garrido, Aurora.Vizcaino, Juan.Andrada, Mario.Piattini}@uclm.es, MiguelJ.Monasor@gmail.com

Abstract—**From recent decades, the phenomenon of globalization is affecting the business model of companies, evolving into a global market that seeks to reduce costs, increase productivity and competitive advantage. The companies engaged in software development are no strangers to this phenomenon, and also being adapted to develop the software in a distributed way at different development teams scattered around the world. This is known as Global Software Development (GSD). This software development paradigm introduces a number of advantages for companies that follow it, but also introduces a number of difficulties and challenges associated with geographical, temporal and socio-cultural distances. One of the major difficulties appears in the Knowledge and Decisions Management as in GSD information comes from many different sources, which makes its management, storage and reuse very complicated. In order to mitigate some of these challenges, we have developed a tool to support the decisions management made in software projects, in the context of global development. Therefore, the system enables the creation, storage, retrieval and transmission of decisions tackled in a software project, carried out in a delocalized way. In addition, the tool allows project managers manage the information of software projects and the most important value is that it also provides techniques to reuse the decisions taken in previous projects into new projects with similar characteristics.**

Keywords—Decision, Knowledge Management, GSD, Software Project.

I. INTRODUCTION

One problem that takes place in Global Software Development (GSD) is about how companies from different locations can create, organize and manage their organizational knowledge. This knowledge is an essential resource as companies can innovate and be competitive. In the companies involved in software development sector, much of its organizational knowledge is composed for decisions made during the development life cycle of software, such as design decisions, decisions of project analysis, etc..., which often are not documented properly or these are documented by a development team in a concrete site but the rest of the sites do not know that this documentation exists. Thus, after it is difficult reuse this knowledge [1, 2].

On the other hand, nowadays there is a tendency to use the method *Rationale* [3-5], which provides the mechanisms to capture and represent the decisions taken at any stage of life cycle of software development projects. Therefore, *Rationale* makes the generation, storage and representation of decisions and their reasoning easier. Because of it we propose to use Rationale in GSD. Moreover, another important aspect to consider in Knowledge Management is that decisions taken and stored by the method *Rationale* in projects, can be retrieved and reused in other projects with similar characteristics. To do this, we propose to use Case-Based Reasoning (CBR) [6, 7], which can provide solutions to different problems based on past experiences.

Therefore, our work focuses on avoiding some of the challenges of knowledge and decision management in the context of GSD, where the information comes from many sources, in different formats and sometimes in different languages. These facts make difficult to know the decisions taken in software projects and reuse of previous lessons learnt [1, 8, 9].

Thus, in this paper a tool designed to make easier to share and take decisions it is described. Moreover, it was attempted to foster the reuse of "lessons learned" that appear in this new form of delocalized development [10, 11]. The need of this tool was born within a research project where we are collaborating: the ORIGIN project (IDI-2010043 (1-5)) in which five companies and two universities are involved. In this project we are researching about methods and tools useful for GSD.

One of the companies which operate in eleven countries, was concerned about its problems documenting and sharing information about its projects, and proposed the creation of this tool. Therefore, DPMTool is the consequence of the requirements proposed by this multinational company whose name is omitted by privacy issues.

The remainder of the paper is structured as follows: In section II, we explain the problems about Knowledge Management in GSD. Then, in section III and IV, the concepts of *Design Rationale* and CBR are respectively described. In V and VI sections, DPMTool, the tool for decisions management in GSD projects is presented. Finally, conclusions and future work are addressed in the final section.

978-0-7695-4788-6/12 $26.00 © 2012 IEEE
DOI 10.1109/ICGSEW.2012.7

IEEE computer society

II. Problems of Knowledge Management in GSD

Knowledge Management (KM) is a process to create, capture, store, search, retrieve, share, transfer and disseminate existing knowledge within an organization, to increase it and prevent its loss and under-utilization [12-17].

The following sections discuss the challenges in each of the four processes that make up the KM.

A. Knowledge Generation

Organizational knowledge creation involves creating new knowledge, or replacing certain files with new knowledge. This new knowledge is obtained though social collaborations and interactions, and though cognitive process itself of each employee [18].

In GSD, this process of creating new knowledge is more complicated than in co-located development as the information comes from many sources and locations, with different formats and in different languages [2, 9].Therefore, it is often difficult to generate new information consistently and avoiding duplication. Another problem that arises is related to the socialization process, which is very convenient for knowledge creation [17]. However, in GSD socialization is more complicated as face-to-face communication is not common and the interactions are via software applications.

B. Storage and retrieval of Knowledge

The storage and recovery of knowledge is closely linked with the above process. As mentioned, in GSD there are multiple sources of knowledge and, at the same time, there are several sites that store and use this knowledge. It is difficult to create a global and common knowledge base to all those organizations involved in GSD, due to the multitude of existing information and formats (videos, documents, business processes, etc...) [19].

Attempting to mitigate the problems of these KM processes, there is a tendency to use the approach *Design Rationale* [20, 21]. It is used to indicate which kind of information is generated and stored, thus providing a concrete type with a common and consistent format. By this way, the teams that work in GSD, can create, store and retrieve this knowledge, for later application. The concept of *Rationale* will be explained in section III.

C. Distribution of Knowledge

One of the main goals of KM is to detect what knowledge can be reused by the companies, so they can be more competitive in the context of globalization. The distribution of knowledge occurs primarily through processes of socialization, where a group of individuals, usually with common interests, share knowledge and try to learn and solve problems together [22].

In GSD, this process is complicated because of the challenges that appear in the communication. The reason of that is that communication tools may not always be available and can prevent or make the transmission of knowledge difficult. In addition, owing to socio-cultural differences, this process of knowledge sharing is more difficult to perform, because such knowledge can be interpreted in a different way in each country, moreover, there are language barriers and misunderstandings can arise [8, 9].

D. Application of Knowledge

There are also difficulties applying the knowledge retrieved, because of all the problems already discussed. We can emphasize the lack of consensus in applying this knowledge, because of the different ways of managing and using knowledge in the different organizations involved in GSD. Moreover, each organization could have its own business processes and management [23].

In order to help in these processes and to reuse lessons learnt and decisions took in other projects, we propose to use CBR which will be explained in Section IV.

III. Design Rationale

Design Rationale is a method for capturing, representing and maintaining records about decisions made by members of a team developing a software project [24, 25]. This method can be applied at any stage of the development life cycle.

Thus, a design rationale is the explicit listing of decisions made during a design process, and the reasons why those decisions were made. Its primary goal is to support designers by providing a means to record and communicate the argumentation and reasoning behind the design process. Therefore, it should include [24, 25].

- the reasons behind a design decision,
- the justification for it,
- the other alternatives considered,
- the trade offs evaluated,
- the argumentation that led to the decision.

This approach of *Rationale* is proposed as a possible solution for problems of knowledge management in GSD, which we discussed above. In this case, the decisions made in the development of a software project are the knowledge which will be generated, stored, transmitted, and reused by using DPMTool. Thus, the different organizations, involved in the global development of software projects, can use the same type of information, stored in a common and consistent way.

In general, the usage of *Rationale* in GSD could bring the following advantages:

- It provides a common mechanism for capturing and storing the decisions made during the software life cycle. This is because it follows the same format and defines the type of information generated.
- It makes the representation of decisions easier as it can be represented graphically and thus, its transmission. It improves the quality of future decisions and the communication between development teams.

- It makes the retrieval and reuse of the decisions easier. The reason is because the type of information stored is well defined.

There are different methods for capturing information in Rationale [26, 27], among which we are going to use **"Dialogue Map"**. It is a graph, where the nodes can represent a question, an idea or an argument for or against these ideas and questions. Thus, you can discuss a decision and add new alternatives and justifications at the time. This kind of graph is based on **IBIS** (**I**ssuer-**B**ased **I**nformation **S**ystems), in which the knowledge is stored and represented in hierarchical way, in the form of decisions/questions and justifications for them. This is the method of representation used in DPMTool. We decided to use it because it is very intuitive and easy to understand and as it is a graphic the problems of communicating with people who does not share the same native language, misunderstanding, could be avoided or decreased.

IV. Case-Based Reasoning

Closely related to the reused of knowledge management is the concept of Case-Based Reasoning (CBR). CBR is the process of solving new problems based on the solutions of similar past problems [28]. Therefore, CBR is a solution to mitigate the problems of the retrieval and application of knowledge in GSD. In our case the knowledge to retrieve and reuse are decisions made in a software project.

Unlike other techniques and artificial intelligence algorithms, such as Rule-Based Reasoning and Genetic Algorithms, CBR is not considered as a technology but rather as a methodology. A methodology indicates how to solve problems from previous solutions stored in the system, but without specifying a particular technology [29].

CBR has been formalized for purposes of computer reasoning as a four-step process [29, 30]:

- **Retrieval** of similar cases to the current problem.
- **Reuse** a proposed solution for a similar case.
- **Revision** of the proposed solution, to better suit the conditions of the new problem.
- **Retention** of the new solution, becoming a new case.

These four steps constitute the CBR methodology. Thus, to solve a new problem, first, you must obtain a description of it, measuring the similarity of the new problem with other previous problems stored in the system. Then, retrieving the solutions to these similar problems and reusing a solution of these cases. Finally, this new problem (with the solution found) is stored in the system, forming a new case.

V. DPMTool

Once described the theoretical basis of this tool we are going to describe it. As it was mentioned above, DPMTool is a tool that supports decision management in GSD projects. Therefore, it should allow the creation, storage, retrieval, transmission and reuse of decisions approached in a software project, developed in a delocalized environment. In addition, it allows managing software projects on which decisions are made. Thus, this tool reduces and eliminates some of the problems that appear in the global development, as it is the lack of control between development teams, as well as the lack of reuse of knowledge acquired by developing previous projects. In order to obtain the requirements of the tool, two resources were used: literature in GSD and interview with one project manager which was involved in the ORIGIN project mentioned previously.

In the following section, we explain how we have achieved these objectives.

VI. Achievemnt of Objectives

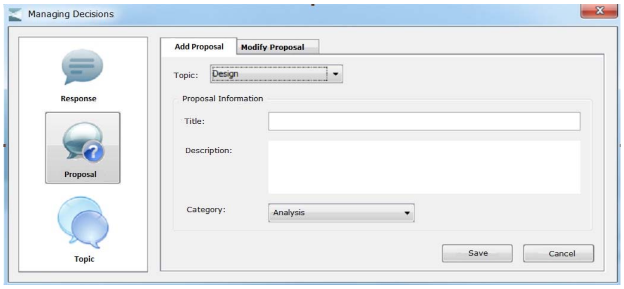
DPMTool has as main goal allowing and making easier decisions management in GSD, such as reusing these decisions for future projects. This section describes how the requirements were implemented.

A. Make easier project management

As discussed in previous sections, in GSD the project management is difficult because the information comes from many sources in different formats. In order to reduce these problems, DPMTool supports the creation and modification of projects by commons forms to all of workers. These forms are about project information and they have several fields to fill in, such as: project's name, description, start and end date, budget, number of code lines, context of project, programming language and estimated hours for its development. By this way, we follow a common structure avoiding errors and misunderstandings. In addition, the tool allows assigning and modifying the users who work in each project. Thus, this feature makes user management easier to project managers because they have to manage only one format of project information and, also this format is very representative because you can know all the project information at a glance.

B. Favour and Make easier decision management

As it is difficult to manage a project in GSD, it is also difficult to know the decisions made in software projects. Thus, software developers need any mechanism to support decision management. DPMTool provides one mechanism to facilitate and encourage decisions management in software projects. As shown in Figure 1, a user can choose between three icons about a decision, following a down-top order: the first one is *topic* in order to model the subject of discussion. Each topic has a title and a description. Another is *proposal*; each topic is composed by proposals. A proposal has a title, a description and a category (this category can be Analysis, Design, Development or Testing). Each proposal is composed by *responses;* that is the meaning of the thirds icon. A *response* has a title, a description and an argument. This argument can be "agree","disagree" or "neutral". In this window, a member can select the first tab to add a new decision, or the second tab to change it.



Topic: Design

Title:

Description:

Category: Analysis

Figure 1. Window for decisions management

After creating or changing a decision, the updates will be reflected visually in the view of decisions (see Figure 2). In this view, in addition to viewing decisions, the user can perform other tasks for managing decisions, such as deleting a decision, including files as attachments, or changing the status of decisions (Accepted or Rejected), we talk about this view below. Only project managers can create and modify a *topic*, while the rest of the team members can add a *proposal* and *response*.

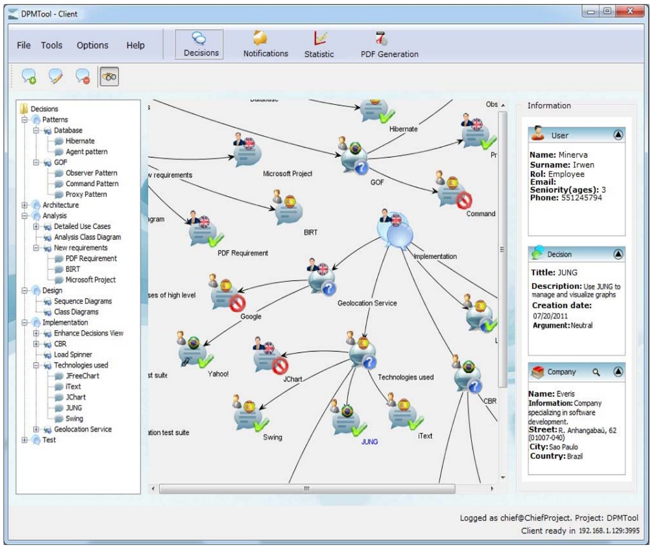
Therefore, all team members can use common forms and structures, minimizing misunderstandings and ambiguities because of information from different sources in different formats. Moreover, we think that visual representation brings less misunderstanding problems than textual representation, mainly when not all the teams' members have the same level of English (language usually used in GSD).

Therefore, the decisions are represented in the same way for all development teams, making the transmission easier and avoiding confusions. Thus, this tool makes the communication between development teams easier because this information is represented in a quick, visual and unambiguous way.

As we can see in Figure 2, the view of decisions is composed by the following items:

- In the left, there is a tree which represents decisions of a software project by hierarchical way. Thus, each Topic has a series of Proposals and these, in turn, have a number of replies.

- In the centre, it displays the same decisions as a graph. Moreover, each icon shows additional information such as the state of a decision: accepted, represented by a "tick" (Figure 3 (a)) or rejected (Figure 3 (b)), documents attachments represented as a paperclick (Figure 3 (c)). The role of the author who created the decision, represented by a man with black suit in the case of being a project manager (Figure 3 (d)) and his country, indicated with the flag of this (Figure 3 (e)). For example, an accepted response with an attachment and created by a Brazilian employee (i.e. not project manager) is represented as shown in Figure 3 (f).



Name: Minerva
Surname: Irwen
Rol: Employee
Email:
Seniority(ages): 3
Phone: 551245794

Tittle: JUNG
Description: Use JUNG to manage and visualize graphs
Creation date: 07/20/2011
Argument: Neutral

Name: Everis
Information: Company specializing in software development.
Street: R. Anhangabaú, 62 (01007-040)
City: Sao Paulo
Country: Brazil

Logged as chief@ChiefProject. Project: DPMTool
Client ready in 192.168.1.129:3995

Figure 2. View of decisions

- In the right, there are several panels, which show detailed information about the author of a decision, his company and his decision. Moreover, the panel with company information displays its geographical position on a map.

Once viewed the decisions, they also can be managed from this viewed. To do this, you should to select a decision from the graph or tree, thus enabled the toolbar buttons that let you add a new decision as a child of the selected decision, modify the decision, or delete it. A decision can only be selected for modifying or deleting by its author, not by another employee. Moreover, clicking the right mouse button on a decision that appears in the graph, it displays a menu that allows you to attach a file, change the status of this decision or remove it.

Finally, if while a user is viewing and managing decisions in this view, and another user is using the tool from another location and make a change on the decisions of the same project, the first user will be notified about the changes that have occurred by another user, in real time.

C. Make easier distribution of new information

In the software development, there are necessary mechanisms to help the communication between workers of a development team. Therefore, this tool has mechanisms for synchronous and asynchronous communication, making communication between users of distributed development teams easier. As for asynchronous communication, the tool generates alerts when there has been any change on the decisions of a project. This alert is automatic and the user can check the alerts in an interface similar to an inbox of e-mails. By other hand, as for synchronous communication, when the client makes any request above management decisions, the server notifies to the rest of connected partners this change. Thus, the clients can see this change visually in real time.

D. Make easier the reuse of information

In the development of projects it is important to reuse the "learned lessons" [10, 11], but this is very difficult in GSD. To solve this problem, DPMTool uses CBR to compare projects and retrieve and reuse decisions from similar projects. As we talked about it above, this technique is focused on solving new problems based on the solutions of similar past problems. For this search of similar past problems, first you should select a search algorithm and set the parameters involved in these algorithms.

Thus, the window to set CBR, it is composed by the following items:

- **Algorithm:** selected algorithm to search in CBR. The algorithm can be *NN* (*Nearest Neighbour*) o *Euclidean Distance*.
- **K:** number of similar projects to be recovered. If it is empty, recovering all completed projects that exist in the system.
- List of attributes that make up a project and are used to calculate the similarity between projects such as budget, number of lines, programming language, etc.
- The type of function applied to compare each pair of attributes among projects. The functions available are:
  - *Equal:* It is used to compare if the value of two attributes of two projects is the same.
  - *Threshold:* This function calculates the similarity of two attributes (numeric or date type) making the difference between the values of those attributes and checking that is smaller than a threshold.
  - *Difference:* It calculates the similarity based on the difference between the values of a pair of attributes, whether numerical, dates or string type. In this last case, the similarity is 1 if the strings are equal, and 0 if not.
  - Weight**:** This value is between 0.0 and 1.0. It is the weight that each attribute is given in the final calculation of the similarity between projects. Thus, a weight equal to 0.0 means that this attribute is ignored in the calculation, while a weight of 1.0 means that the attribute is considered very important to calculate the similarity.

After that, DPMTool displays the project data recovered, employees who work on it, the decisions made in the project and the percentage of similarity respect to the draft that began with the search. Moreover, it is possible to navigate between the different projects found. In addition, the decisions can be filtered by status (Open, Accepted or Rejected), and it is also possible to store project information as a PDF or XML file. Thus, information and project decisions can be reused in an initial project similar to this one.

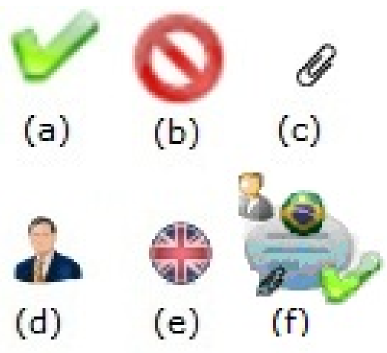


Figure 3. Icons of DPMTool

E. Make easier sharing knowledge

Because each co-worker is dislocated in a different place, it is necessary for workers to be able to share their information between them, and this information should be in a common format. This tool can export information above decisions and projects to XML files. In addition, it can generate PDF reports and statistical graphics. Thus, project managers can keep track on the project and all its associated information. In addition, this tool helps to project managers because it can generate several reports and graphs in a simple and visual way.

VII. Conclusions And Future Work

This paper describes a tool which has been implemented to solve some of the typical problems that take place in GSD. The main goal when designing this tool was to support the decisions management made in software development. Thus, DPMTool can create, store, retrieve and transmit decisions tackled in a software project. Other features have also been added to improve communication and coordination.

As future work we are planning to add new features in the description of the cases of CBR. These features will be related to the settings of the project, for instance time overlap among the different sites, degree of cultural difference among team members and level of knowledge about common language. In order to make a quantitative estimation of this factors the proposal by Aranda et al. will be used [31].

Acknowlegment

This work has been funded by the PEGASO/MAGO project (Ministerio de Ciencia e Innovación MICINN, TIN2009-13718-C02-01) and co-funded by FEDER. It is also supported by ENGLOBAS (PII2I09-0147-8235) and GLOBALIA (PEII11-0291-5274), funded by Consejería de Educación y Ciencia (Junta de Comunidades de Castilla-La Mancha) and co-funded by FEDER (Spain), as well as, ORIGIN (IDI-2010043 (1-5)) funded by CDTI and FEDER.

REFERENCES

[1] I. Richardson*, et al.*, "Knowledge Management in the Global Software Engineering Environment," presented at the Fourth IEEE International Conference on Global Software Engineering Limerick, Ireland, 2009.

[2] A. Nour, "Architectural Knowledge Management in Global Software Development: A Review," presented at the Global Software Engineering (ICGSE), 2010 5th IEEE International Conference, Princeton, NJ, USA, 2010.

[3] J. Horner and M. E. Atwood, "Effective Design Rationale: Understanding the Barriers," *Rationale Management in Software Engineering,* pp. 73-90, 2006.

[4] A. P. J. Jarczyk*, et al.*, "Design rationale for software engineering: a survey," presented at the System Sciences, 1992. Proceedings of the Twenty-Fifth Hawaii International Conference on Global Software Engineering,, 1992.

[5] X. Wang and G. Xiong, "Design rationale as part of corporate technical memory," presented at the Systems, Man, and Cybernetics, 2001 IEEE International Conference on Global Software Engineering, 2001.

[6] A. Agnar and P. Enric, "Case-based reasoning : Foundational issues, methodological variations, and system approaches," ed, 1994.

[7] A. Ram and N. Wiratunga, "Case-Based Reasoning Research and Development," in *19th International Conference on Case-Based Reasoning, ICBR 2011*, London, UK, 2011.

[8] V. Clerc*, et al.*, "The Usefulness of Architectural Knowledge Management Practices in GSD," presented at the International Conference on Global Software Engineering, Limerick, Ireland, 2009.

[9] V. Clerc, "Do Architectural Knowledge Product Measures Make a Difference in GSD?," presented at the ICGSE 2009. Fourth IEEE International Conference on Global Software Engineering, , Limerick, Ireland, 2009.

[10] S. Krishna*, et al.*, "Managing cross-cultural issues in global software outsourcing," *Communications of the ACM - Human-computer etiquette* vol. 47, pp. 62-66, April, 2004 2004.

[11] A. Aurum and C. Wohlin, Eds., *Engineering and Managing Software Requirements*. Springer Verlag, 2005, p.^pp. Pages.

[12] K. A. Kanagasabapathy*, et al.*, "Empirical investigation of critical success factor and knowledge management structure for successful implementation of knowledge management system: A case study in process industry," 2003.

[13] M. Bomemann*, et al.* (2003). *Illustrated Guide to Knowledge Management*.

[14] I. Hovland, "Knowledge management and organisational learning: An international development perspective," presented at the Overseas Development Institute Working Paper No. 224, London, UK., 2003.

[15] S. Beecham*, et al.*, "Crafting a Global Teaming Model for Architectural Knowledge," pp. 55-63.

[16] R. Jabangwe and D. Smite, "Decision Support for Offshore Insourcing Software Development," presented at the IEEE Sixth International Conference on Global Software Engineering Workshop, Helsinki, Finland, 2011.

[17] I. Nonaka and G. von Krogh, "Perspective-Tacit Knowledge and Knowledge Conversion: Controversy and Advancement in Organizational Knowledge Creation Theory," *Organization Science,* vol. 20, pp. 635-652, May/June 2009 2009.

[18] M. Alavi and D. E. Leidner, "Review: Knowledge management and knowledge management systems: Conceptual foundations and research issues," *MIS quarterly,* pp. 107–136, 2001.

[19] M. A. Babar, "A Framework for Supporting the Software Architecture Evaluation Process in Global Software Development," in *ICGSE 2009. Fourth IEEE International Conference on Global Software Engineering*, 2009, pp. 93-102.

[20] L. Peng*, et al.*, "Requirements Reasoning for Distributed Requirements Analysis Using Semantic Wiki," presented at the Global Software Engineering, 2009. ICGSE 2009. Fourth IEEE International Conference on, 2009.

[21] C. Lopez*, et al.*, "NDR Ontology: Sharing and Reusing NFR and Design Rationale Knowledge," presented at the MARK '08. First International Workshop on Managing Requirements Knowledge, 2008.

[22] E. Wenger, *Communities of practice: Learning, meaning, and identity*: Cambridge University Press, 1998.

[23] F. García*, et al.*, "Process Management Tools," *IEEE Softw.,* vol. 28, pp. 15-18, 2011.

[24] A. H. Dutoit*, et al.*, *Rationale management in software engineering*: Springer, 2006.

[25] D. McKerlie and A. MacLean, "Reasoning with Design Rationale: practical experience with design space analysis," *Design Studies,* vol. 15, pp. 214-226, 1994.

[26] J. Lee, "Design rationale systems: understanding the issues," *IEEE Expert,* vol. 12, 1997.

[27] Z. Peisi*, et al.*, "History knowledge management in lifecycle of product development and its implementation," presented at the Proceedings of the Ninth International Conference on Computer Supported Cooperative Work in Design, 2005.

[28] D. W. Aha*, et al.*, "Case-based reasoning commentaries: introduction," *The Knowledge Engineering Review,* vol. 20, pp. 201-202, 2005.

[29] I. Watson, "CBR is a methodology not a technology.," *Research & Development in Expert Systems XV,* pp. 213–223, 1998.

[30] A. Aamodt*, et al.*, "Retrieval, reuse, revision and retention in case-based reasoning," 2006.

[31] G. Aranda*, et al.*, "A framework to improve communication during the requirements elicitation process in GSD projects," *Requirements Engineering,* vol. 15, pp. 397-417 November, 2010 2010.