

RE II: RE Tools

Requirements-Centred Collaboration

Christian Anastasiu, Samuel Liniger, Fabian Hofstetter



Literature

stakeholders in RE.....

Wiki-Based Stakeholder Participation in Requirements Engineering

Björn Decker, Eric Wan, Hing Hech, Pascal Janbert, and Marco Bloch,
Fraunhofer Institute for Experimental Software Engineering

Wikis offer a flexible platform for asynchronous collaborative support of active stakeholder participation in requirements engineering.

Research has shown that, in projects with high uncertainty, stakeholders' participation improves the quality of requirements products.¹ Particularly in distributed environments, participative requirements engineering needs a platform that can support effective, efficient collaboration among an often large number of diverse stakeholders. The platform must account for the following challenges concerning stakeholders:

- **different perspectives** on the system under development;
- different **backgrounds**, which can cause communication problems;
- different **objectives**, which influence views on the requirements;
- different **abilities to express requirements** and document them using a selected platform; and
- different **involvements**—for example, some stakeholders are entitled to make decisions and others aren't.

It's important to address these challenges by **enabling communication** and **decision making**. In particular, project participants need a way to detect misunderstandings and conflicts and solve them as early as possible. This communication can be synchronous—through face-to-

face meetings and videoconferencing, for example—or asynchronous, which is our focus here. Requirements elicitation and documentation are complex activities. So, not only the requirements themselves but also the people involved and the means for managing the requirements will evolve during the project. For example, it might be necessary to add RE personnel, to document templates, or to change the requirements classification scheme.

In summary, especially in participative RE, the underlying platform as well as the RE approach must be flexible.

As the well-known Wikipedia shows, wikis provide a flexible platform for asynchronous collaboration to create content in general.² In this article, we investigate how to adapt this approach to support active stakeholder participation in RE. We include a document structure for

global software development.....

Enabling Collaboration in Distributed Requirements Management

Vibha Sinha and Bikram Sengupta, IBM India Research Lab
Sudish Chandra, IBM TJ Watson Research Center

The Exact tool can support global software development teams in collaborating on requirements management.

Requirements management, one of the most collaboration-intensive activities in software development, presents significant difficulties when stakeholders are distributed, as in today's global projects. Because of inadequate social contact, geographically distributed practitioners without appropriate tool support have trouble gaining a consistent understanding of requirements or managing requirement changes. We can alleviate many of these difficulties by integrating

collaboration support in practitioners' work environments.

With these needs in mind, we developed a new collaborative tool named **Exact (Enterprise-based global requirements tool)** for distributed requirements management. From a tools perspective, we wanted to explore how to weave in collaboration services for distributed requirements management. We were equally interested in how the practitioners with whom we had interacted to derive our scenarios would react to such a tool—in particular, which features they'd most appreciate. Here, we present Exact's design motivations, a preliminary evaluation of its usability, and suggested directions for further enhancement.

The problem

For more than a year, we interacted extensively with IBM practitioners engaged in dis-

tributed development to understand and propose solutions for their pain points. We spoke to approximately 50 practitioners in different roles—both onsite people in the US and the Netherlands and remote team members based in India—across 14 projects. For data collection, we primarily used semistructured interviews conducted through face-to-face meetings and teleconferences.

The practitioners reported a wide range of issues, from the well-known difficulties of cultural and time-zone differences to challenges related to data privacy, application knowledge migration, project management, and quality metrics. These findings, and a research agenda for global software development that we proposed based on them, are documented elsewhere.³ Other researchers have also reported some of these challenges.⁴

Communicating and managing require-

Wiki-Based Stakeholder Participation in Requirements Engineering - Decker et al. (2007)

Enabling Collaboration in Distributed Requirements Management - Sinha, Sengupta & Chandra (2006)

Context

Collaborative Software

1. Level
Electronic
communication
tools

2. Level
Electronic
conferencing
tools

3. Level
Collaborative
management
(coordination)
tools

Definition

- ◆ Collaborative Software

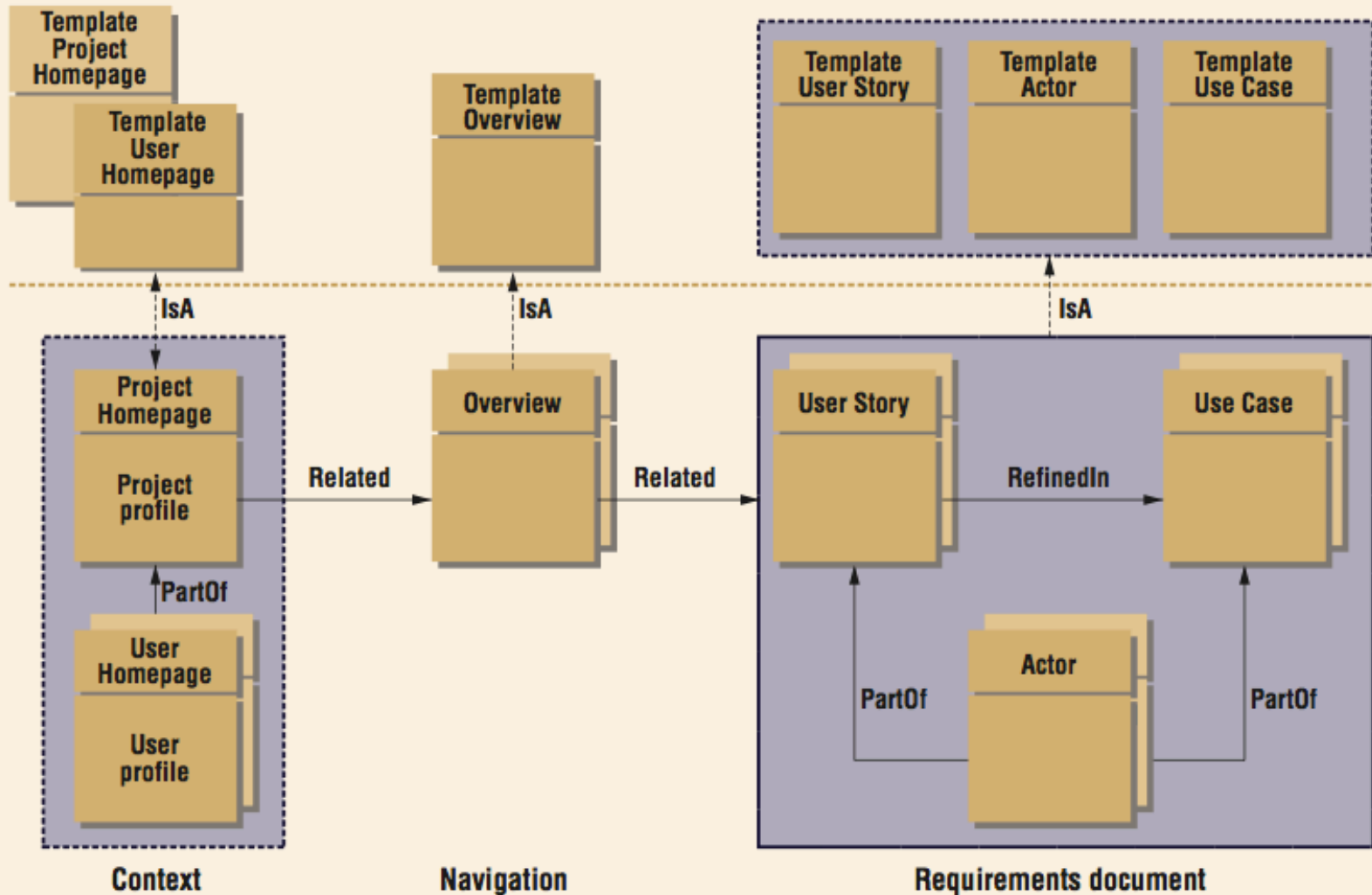
- ◆ **Collaborative software** or **groupware** is an application software designed to help people involved in a common task achieve goals. The design intent of collaborative software is to transform the way documents and rich media are shared to enable more effective team collaboration. [1]

Wiki-based Stakeholder participation

- ◆ Stakeholders in RE projects:
 - ◆ Different perspectives
 - ◆ Different backgrounds
 - ◆ Different objectives
 - ◆ Different abilities to express their needs
 - ◆ Different involvements



Wiki: Document Structure



Wiki: Challenges

- 💧 Exporting content
- 💧 Intelligent versioning
- 💧 Intelligent linking
- 💧 Offline working



Wiki

Pro

- 💧 Collaboration Support
- 💧 Grouping requirements
- 💧 Versioning of requirements
- 💧 Low cost
- 💧 Also usable in other development phases

Con

- 💧 Missing replication
- 💧 Increasing complexity of wiki-syntax to add metadata

Collaboration in Distributed Requirements Management

- ◆ Problems of Distributed Development
 - ◆ Cultural
 - ◆ Time-zone
 - ◆ data privacy etc.
- ◆ Problem Distributed Requirements Management:
 - ◆ Frequent context switches
 - ◆ information fragmentation across several media
 - ◆ → lack of common understanding

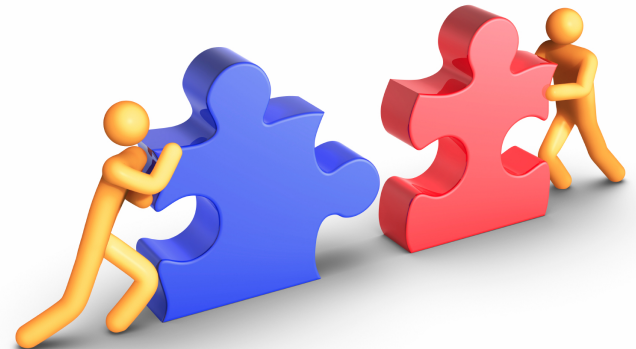
Problems / Goals

- ◆ Communication & discussion
- ◆ Managing changes
- ◆ Managing project knowledge



Collaboration

- ◆ Informal collaboration services
- ◆ Structured collaboration services
- ◆ Awareness features
- ◆ Knowledge management services

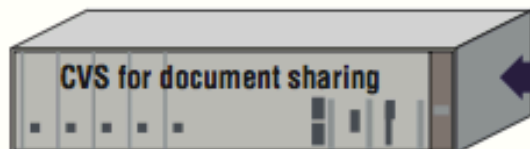
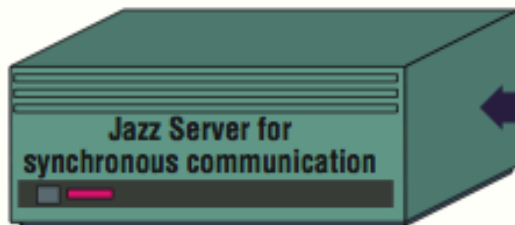


EGRET

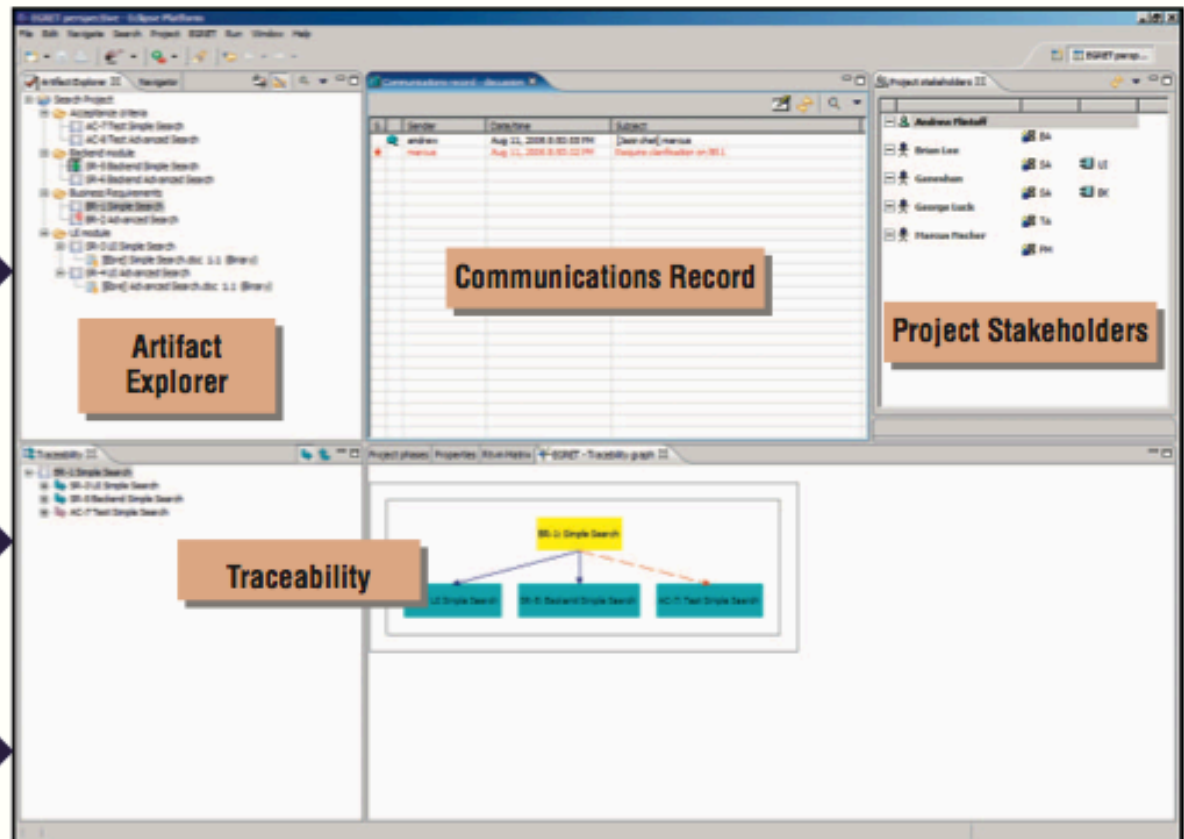
Server side

MySQL database

- Requirements data
 - Name, text, hierarchy
 - Traceability
- Communication data
 - Discussions, change notifications, requests
- Change log
- People and project information
 - People, role, responsibility, project phases, modules



Client side



EGRET

- ◆ Communication & discussion
 - ◆ Synchronous & Asynchronous contextual collaboration
- ◆ Managing changes
 - ◆ Requirements owner & information distribution mechanisms
- ◆ Promoting awareness
 - ◆ Visual representation of state of requirement
- ◆ Managing project knowledge
 - ◆ Comprehensive search & visualization of traceability

Take Home Messages

- ◆ Wikis can help you to manage Stakeholders and increase Stakeholder Participation in the Requirements Elicitation process
- ◆ Deep integration of collaboration support in requirements management can aid distributed development teams.



Discussion

- ◆ Do you think that an Eclipse-based tool is appropriate for requirements management?
- ◆ Do you think that also non-technical Stakeholders can express themselves in a Wiki?

