

# Software Architecture as a Means of Communication in a Globally Distributed Software Development Context

Richard Berntsson Svensson<sup>1</sup>, Aybüke Aurum<sup>2</sup>, Barbara Paech<sup>3</sup>,  
Tony Gorschek<sup>4</sup>, and Devesh Sharma<sup>2</sup>

<sup>1</sup> Department of Computer Science School of Lund University Lund, Sweden  
richard.berntsson\_svensson@cs.lth.se

<sup>2</sup> School of Information Systems, Technology and Management,  
University of New South Wales Sydney, Australia  
aybuke@unsw.edu.au, d.sharma@unswalumni.com

<sup>3</sup> Institut für Informatik University of Heidelberg, Heidelberg, Germany  
paech@informatik.uni-heidelberg.de

<sup>4</sup> School of Computing Blekinge Institute of Technology Karlskrona, Sweden  
tony.gorschek@bth.se

**Abstract.** The management and coordination of globally distributed development poses many new challenges, including compensating for informal implicit communication, which is aggravated by heterogeneous social and engineering traditions between development sites. Although much research has gone into identifying challenges and working with practical solutions, such as tools for communication, little research has focused on comparing communication mechanisms in terms of their ability to provide large volumes of rich information in a timely manner. Data was collected through in-depth interviews with eleven practitioners and twenty-eight responses through a web-based questionnaire from three product lines at an international software development organization. This paper assesses the relative importance of ten commonly used communication mechanisms and practices across local and global development sites. The results clearly indicate that some communication mechanisms are more important than others in providing large volumes of rich information in a timely manner. The prevalence of architecture in providing rich information in large volumes for both local and global communication can be clearly observed.

**Keywords:** Global Software Engineering, Communication, Case Study, Software Product Lines, Software Architecture, Product Management.

## 1 Introduction

Software product development and software product management have emerged as ways of developing software as a product for the mass-market [27], rather than for a specific customer [26]. Software product managers steer software product development towards a beneficial direction for the company by selecting requirements for the coming releases of a product and creating business objectives [19], while the

development teams formalize the product's functionality and assure its quality [17] to increase the likelihood of market success. Although a software product's functionality and quality are important for the success of the product, the collaboration between software product management and the development team is crucial for product success [8]; however, the collaboration requires a handle of communication and coordination challenges [9].

However, large-scale software development can be complicated, expensive and unpredictable [3]. For software companies to succeed in the global markets of software-intensive products, shortened cycle time of product development and improved product quality are essential. To achieve shortened cycle time and improved quality, two internal strategies can be used, namely software product lines (SPL) and global software engineering (GSE). Potential advantages of GSE include "round-the-clock" development, access to global markets, and reduced development time and cost [11]. However, SPL and GSE are further accelerating the complexity of software product development [3]. Furthermore, when a product line is adapted to GSE, processes, tools, and organizational structure changes [2], and has significantly more difficulties to implement the necessary coordination [3].

One key factor of the software product development process is the process of communication, coordination, and collaboration [4, 6, 12, 25]. However, it is not only the formal communication process that impacts the development. Several studies [7, 13, 16] observe that developers rely on informal and ad-hoc communication. Lack of, or problems in the informal communication channels may lead to increased development time [16]. Communication issues in GSE have been addressed by other studies; see e.g. [6, 11, 16], while communication in new product development has been addressed by, e.g., [8]. However, none of these have focused on comparing local peer-to-peer (e.g. face to face meetings), long-distance peer-to-peer (e.g. electronic chat, including instant messaging) and technical (e.g. architecture) communication tools in GSE, for their ability to provide information in a timely manner, with richness and with large volumes of information.

This paper presents the result of an empirical study that includes data collected through in-depth interviews with eleven practitioners and twenty-eight responses through a web-based questionnaire from three product lines in one international software development organization operating in over one hundred countries around the world. The case organization uses SPL in a GSE context. The main objective of this study is to assess the relative importance of ten commonly used communication mechanisms and practices from three different aspects, for their ability to transmit information quickly, transmit rich information, and to transmit large volumes of information across local and global development sites. The study incorporates two main perspectives with regards to communication mechanisms through the study of local and global development sites.

The remainder of this paper is organized as follows. In Section 2, background and related work are presented. The case organization is described in Section 3, and Section 4 describes the research methodology. Section 5 presents the results, while the results are discussed in Section 6. Validity threats are discussed in Section 7, and Section 6 gives a summary of the main conclusions.

## 2 Background

In SPL, software development is coordinated between teams in several different ways, during the front-end of the development, during the development, and during the end of the development [3]. During the front-end of the development, roadmaps need to be shared, discussed, and agreed upon. During the development phase, new components and interfaces need to be coordinated, while in the end of the development phase integration needs to be coordinated.

Over the last few decades, the software industry has been exposed to a steady and irreversible trend towards the globalization of business [25]. The global expansion of software organizations has made companies aware of the potential advantages that GSE has to offer, such as capitalization of global resources pools, “round-the-clock” development and access to global markets, reduced development time and cost [11]. On the other hand, these benefits are not clear-cut and should not be taken-for-granted [5]. Ó Conchuir et al., concluded that overall development costs might not be reduced due to the introduction of higher managerial complexities [5]. Furthermore, “round-the-clock” development seemed unrealistic, and companies may prefer to modularize work [5]. In addition, there are a number of challenges introduced by GSE, such as communication, coordination, and control of the development process [5].

There exists no shortage of studies relating to communication in a GSE context. Herbsleb and Grinter showed the importance of informal communication and the difficulties in communicating across global sites [10]. In fact, a major challenge in GSE is the lack of informal communication [7]. Herbsleb and Mockus further found that cross-site work takes longer than same-site work [11]. In addition, cross-site teams seem to have relatively little understanding of the overall context compared to same-site teams [11].

As for same-site development, the communication process, particularly informal communication [11], is an important factor for success in GSE [4, 6, 12]. According to Curtis et al., communication barriers in cross-site teams can be mitigated by an architect who acts as a boundary spanner between the teams [7]. Most, if not all, stakeholders in the software development can use software architecture as a basis for mutual understanding, negotiation, consensus, and communication. Bass et al. point out that the architecture provides a common language in which different concerns can be expressed, negotiated, and resolved at a level that is intellectually manageable, even for complex systems [1]. Ovaska et al., found that in a multi-site development environment, developers coordinate their work through the use of well-defined interfaces and an appropriate architecture description [16]. Using such an approach also means that components can be developed separately and the impact of distance, language and culture are minimized.

Furthermore, it is important how communication tools are utilized. Niinimäki and Lassenius conducted a multiple case study to investigate how instant messaging was used [15]. The results show that instant messaging was used to communicate simple questions and clarifications, as well as technical decisions and solutions. Šmite found that the communication channels used by a GSE organization were mainly email and telephone [24]. Very seldom was communication conducted through meeting in

person, while videoconferences were never used. In McDonough et al., eight different communication tools were assessed based on speed, richness, and volume [14]. The results show that e-mail and company databases in the organization were the fastest communication tools; face-to-face meetings provided the richest information, while email and databases were viewed as providing a large volume of information.

### 3 Case Description

Data was collected from an international software development organization operating in over 100 countries around the world. The organization primarily specializes in database management systems, enterprise resource planning, customer relationship management and other industry-tailored products targeting government, finance and healthcare sectors. Based in the USA, it employs over fifty thousand workers around the world, with more than 25% of its workforce involved in software development. For the purposes of confidentiality, this organization will be simply referred to as “the company” throughout this paper. All of the company’s products stem from its four major software product lines (SPL). The company has a few teams actively involved in the development of its software in Australia, and a larger research and development base in India. The company had four individual SPL at the time this research was conducted. However, development in Australia was only conducted for three of the SPL. Hence, only three SPL could be examined in this study.

*Product line A (PLA)* comprises enterprise resource planning, supply chain management, customer relationship management, human resources and industry-specific applications targeting banking and healthcare. This is the company’s original product line and has been undergoing iterative development for over ten years. It now boasts a large product mix, but has an aging core asset base with slow evolution of components and architecture. The products in PLA are currently in their 12th major release.

*Product line B (PLB)* consists of a collection of products offering solutions for human resource management, customer relationship management, manufacturing, and student administration software for large corporations and government sectors. The company acquired this product line in 2005 through a takeover of its parent organization. PLB has a relatively well maintained core asset base, and has a proprietary integrated development environment which forces developers to reuse core assets. The products in PLB are currently in their ninth major release, and its architecture is built around the company’s own proprietary development platform.

*Product line C (PLC)* is the company’s latest collection of products aimed at unifying the best-of-business capabilities offered by its applications and other product lines. Through the use of an open, service oriented architecture, PLC is used as a standards-based technology blueprint that enables effective, predictable business process changes through standards-based integration of applications developed as web services. Developers and managers in PLC follow strict standards that do not allow for the duplication of core assets, and encourage evolution of existing assets. To date, most PLC products are still undergoing development and have yet to be released.

## 4 Methodology

The main objective of this research is to examine commonly used communication mechanisms and practices across local and global development sites. The research questions, in Table 1, provide the focus for the empirical investigation. RQ1 is a macro question that is broken down into three discrete sub-questions, each addressing the separate aspects of timeliness, speed, information richness and scalability of information load. To address the research questions an exploratory case study was undertaken. This study was built on semi-structured interviews [20] with a high degree of discussion between the interviewer and the interviewee, complemented with a questionnaire. This study was conducted purely from the perspective of the software organization, i.e. customers were not involved in this research. The study consisted of two steps, described in the following section.

**Table 1.** Research questions

<i>Research Questions</i>
<b>RQ1:</b> What communication mechanisms are central to a distributed SPL environment in order to provide large volumes of rich information in a timely manner?
<b>RQ1.1:</b> What communication mechanisms are central to a distributed SPL environment in terms of speed?
<b>RQ1.2:</b> What communication mechanisms are central to a distributed SPL environment in terms of providing rich information?
<b>RQ1.3:</b> What communication mechanisms are central to a distributed SPL environment in terms of providing large volumes of information?

### 4.1 Step 1: Interview Study

*Planning:* The first part of the study involved brainstorming and planning to design the study and to identify different areas of interest. The contact person at the company assisted in the identification of appropriate participants to be interviewed. The interview instrument was designed with respect to the different areas of interest i.e. company and personal background details, product and SPL background, requirements engineering process, architecture design, and impact of GSE. In addition, the interview instrument examined coordination issues in a global setting, and the communication mechanisms used by the company to communicate, both locally and globally. The interview instrument is available in [21]. To test the interview instrument, pilot interviews were carried out. Some questions were clarified and the structure of the interview instrument was improved before interviewing proceeded.

*Data Collection:* The study involved eleven interview participants in the roles of product managers (PM) and development managers (DM). The distribution of participants among the three SPL are displayed in Table 2. All interviews were attended by one interviewee and one interviewer. Each interview took approximately an hour. The interviews from Australia were recorded, while notes were taken at interviews involving Indian participants. Transcripts of all interviews were made in order to facilitate and improve the analysis process.

*Analysis:* An interpretive analysis of the interview data was conducted to address the research objective [23]. Content analysis of the transcribed interview data was also conducted using the Leximancer content analysis software. The interviewer examined the transcripts from different perspectives. In addition, another researcher, who did not attend the interviews, also analyzed the transcripts to enhance validity. Preliminary results from Step 1 are presented in [22].

**Table 2.** Distribution of Participants in Interviews

<i>Interviews</i>		
Country	Australia	India
PLA	4 (2 PM, 2 DM)	0
PLB	4 (3 PM, 1 DM)	0
PLC	1 (1 PM)	2 (1 PM, 1 DM)
<b>Total</b>	9	2

**Table 3.** Distribution of Participants in Questionnaire

<i>Questionnaire</i>		
Country	Australia	India
PLA	6	7
PLB	2	6
PLC	2	5
<b>Total</b>	10	18

## 4.2 Step 2: Questionnaire

*Planning:* The aim of the questionnaire was to understand the importance of different communication practices used in local and global communication by development teams, and to confirm the findings from the interviews. The questionnaire<sup>1</sup> was adopted from McDonough et al. who systematically studied eight different communication mechanisms, including phone calls, fax, e-mail, teleconference, face-to-face meetings, mail, company databases, and videoconferencing, in terms of their unique information transmission capabilities [14]. The communication mechanisms were rated with respect to their ability to transmit information quickly (speed), to transmit rich information (richness) and to transmit large volumes of information (volume). The questionnaire was modified by replacing the original communication mechanisms with ten communication practices that were commonly used in the company. The communication practices were identified through the study of process and project management documentation at the company and the initial interviews as follows (for explanations of the ten communication mechanisms see footnote 1): software architecture, code walkthroughs, visiting engineer, regular meetings, change management processes, discussion forums, electronic chat, face-to-face communication and process walkthroughs. The participants rated the communication mechanisms for their ability to transmit information quickly (speed), to transmit rich information (richness) and to transmit large volumes of information (volume). The rating mechanism used in the original questionnaire was also modified from independently ranking communication mechanisms, to the ranking of relative importance of different mechanisms when compared to each other. Participants were required to compare the different communication mechanisms against each other and attach a weighting to the importance of different communication mechanisms. This was enabled through the distribution of a

<sup>1</sup> [http://serg.cs.lth.se/research/experiment\\_packages/GSE](http://serg.cs.lth.se/research/experiment_packages/GSE)

thousand ‘points’ across the ten different communication mechanisms. To test the questionnaire, two pilot studies were carried out. Some questions were clarified and improved.

*Data Collection:* The questionnaire targeted employees involved in the development of products within each of the studied SPL. The company contact assisted with the identification of appropriate software developers within Australia to participate in the questionnaire. In order to obtain a balanced perspective on the use of communication practices in the global environment, the questionnaire was also distributed to the Indian branch of the company. Participants in India were selected through the identification of Indian counterparts of participants in Australia.

In total, 28 of 53 participants completed the questionnaire, yielding a response rate of 53%. Participants were mainly project and product managers, application architects and application engineers. The distribution of participants among the three SPL, and between Australia and India, are displayed in Table 3.

*Final Analysis of All Data:* The questionnaire results were analyzed for each PL individually. The importance of each criterion, at both local and global level, was analyzed by summing all the points for the respective criterion, followed by normalizing the result for each criterion to a percentage. Since a comprehensive view of the complete data set was sought, the data from the first part of the study was re-analyzed together with the data from the questionnaire.

## 5 Results and Analysis

The following sub-sections present and discuss one research question each, corresponding to the research questions in Table 1.

### 5.1 Most Suitable Communication Mechanism (RQ1)

This section examines the relative importance of the most suitable communication mechanisms to deliver large volumes of rich information in an effective timeframe.

The different communication mechanisms are divided into three main categories: (1) *local peer-to-peer*, which includes face-to-face, regular meetings, and visiting engineer, (2) *long distance peer-to-peer*, which includes electronic chat and forums, and (3) *technical*, which includes architecture, code walkthrough, process walk through, progress report, and change management. We calculated the average relative importance of speed, richness, and volume for each of the ten communication mechanisms. In addition, to understand what communication mechanism provides large volumes of rich information in a timely manner, we summed all points from all three aspects at local and global levels, and calculated the average relative importance of the combined total sum for each SPL individually. The result is shown in Table 4.

The results clearly indicate that some communication mechanisms are more important than others in local and global site communication. It is worth noting the difference in the top communication mechanisms when compared for their ability to provide information in a timely manner, with richness and with large volumes of information.

**Table 4.** Local vs. global distributed communication mechanisms

	PLA		PLB		PLC	
	<i>Local</i>	<i>Global</i>	<i>Local</i>	<i>Global</i>	<i>Local</i>	<i>Global</i>
<b>Technical communication mechanisms</b>						
Architecture	12.13%	<b>17.53%</b>	<b>15.90%</b>	<b>16.73%</b>	11.93%	<b>15.97%</b>
Change Management	6.87%	7.43%	7.20%	5.47%	7.20%	6.97%
Code Walkthrough	11.03%	9.57%	9.00%	7.27%	9.13%	7.70%
Process Walkthrough	7.23%	6.80%	13.13%	8.30%	7.57%	11.53%
Progress Report	7.00%	8.53%	4.93%	6.10%	6.63%	8.10%
<b>Local peer-to-peer communication mechanisms</b>						
Face-To-Face	<b>14.80%</b>	8.70%	<b>15.40%</b>	8.50%	<b>14.77%</b>	7.20%
Regular Meeting	10.80%	8.50%	7.07%	9.53%	11.20%	10.83%
Visiting Engineer	10.50%	14.00%	7.20%	16.17%	9.73%	12.63%
<b>Long distance peer-to-peer communication mechanisms</b>						
Forums	8.53%	5.80%	7.60%	5.00%	11.50%	13.47%
Electronic Chat	11.10%	13.17%	12.57%	<b>16.90%</b>	10.43%	5.60%

For local communications, face-to-face was seen by participants from PLA as a communication method that delivered large volumes of rich information in an effective timeframe in PLA. Other important communication mechanisms were architecture, followed by electronic chat. The result (the order of communication mechanisms) for PLC was exactly the same as for PLA, with the difference that forums replaced electronic chat as the third preferred communication method. For PLB, the result was similar; architecture was seen as the most suitable criterion, followed by face-to-face and process walkthrough. One major difference was that participants from PLC perceived all technical communication mechanisms, with the exception of architecture, as providing small volumes of less rich information in a relatively slow manner. This result was not consistent with the result from PLA and PLB, where, for example, participants viewed code walkthrough as a better alternative to regular meetings and visiting engineers.

Looking at global communication sites, architecture and the presence of a visiting engineer were perceived as delivering large volumes of rich information in an effective timeframe. In PLA and PLB, electronic chat was viewed as almost equally important, and again, participants from PLC preferred forums over electronic chat. It is interesting to note, for global communication, the participants from all three SPL prefer either electronic chat or forums, never both. The less preferred one was seen as the least effective communication method that delivered the smallest volume of less rich information. One interesting finding was the view of technical communication mechanisms in PLB. All of the technical communication mechanisms, except for architecture, were perceived as the least suitable communication methods, together with forums. This result is not consistent with PLA and PLC. It is surprising to note that participants from PLC viewed electronic chat as delivering relatively large volumes of rich information in an effective timeframe for local site communication, but not in communication with offshore teams.

Despite participants from PLA considering architecture to be the most suitable communication mechanism to deliver large volumes of rich information in an effective timeframe, the interviews revealed that the used mechanism when communicating with offshore teams were conference calls, group meetings, review tools and documentation. In addition, the main constraints imposed in communication with offshore teams in PLA participants were the inability to have in-depth discussions and the lack of body language.

Similar to PLA participants, PLC participants considered architecture as the most suitable communication mechanism for offshore communication. Despite electronic chat being considered by PLC participants as the least suitable mechanism for communication, several interviewees' explained that this was one of the tools used in PLC. Other communication mechanisms used in PLC were telephone and web conferences, documents, and electronic mails. Furthermore, according to PLC participants, the main constraint of offshore communication was the lack of face-to-face meetings. The lack of face-to-face meetings may have an impact on the decision-making process because *"when people do not meet face-to-face, they face a lack of understanding of capability and abilities"*.

PLB participants had a different view than PLA and PLC participants on which communication mechanism was the most suitable for offshore communication. For PLB participants, electronic chat was the mechanism that provided large volumes of rich information in a timely manner. However, electronic chat was not used in PLB, instead, offshore communication involved web and telephone conferences and documents. According to one interviewee in PLB, *"nothing beats face-to-face meetings. Meetings are conducted a lot easier, and in a more understanding manner when done face-to-face"*. However, PLB participants only considered a visiting engineer as the third most suitable communication mechanism.

For all three SPLs, the perceived most suitable communication mechanism was not used in practice. Instead, less effective mechanisms were used when communicating with offshore teams. No further elaboration was given on the topic by participants.

## 5.2 Communication in a Timely Manner (RQ1.1)

In analyzing Research Question 1.1, this section examines which communication mechanism provides information in a timely manner. Among PLA participants, for local site communication, all *local peer-to-peer* communication mechanisms and electronic chat were perceived as the fastest methods to distribute information (see Table 5), meaning that peer-to-peer communication mechanisms were the perceived as faster than technical communication mechanisms. For PLB participants, there was a mix of mechanisms that were perceived as the fastest. The two quickest communication mechanisms were face-to-face and electronic chat, however code walkthroughs and process walkthroughs were also perceived as relatively quick. The result for PLC is similar to PLA and PLB where face-to-face communication was the fastest communication mechanisms. In addition, regular meetings, electronic chat, and code walkthroughs were perceived as quicker than other communication mechanisms in distributing information.

For local site communications, all three SPL viewed face-to-face and electronic chat communication as the fastest methods to distribute information. In PLA, all *local peer-to-peer* communication mechanisms and electronic chat were perceived to be faster than any of the technical communication mechanisms. Unlike PLA, both PLB and PLC participants perceived the *technical* criterion code walkthrough to be faster than a visiting engineer. In addition, PLB participants perceived process walkthrough to be quicker than regular meetings and visiting engineer.

The results for communication between global sites evidently show that distributed teams, in terms of fast methods to distribute information, rely heavily upon a mix of *local* and *long distance peer-to-peer* communication, such as visiting engineers, electronic chat, and forums. The only other communication criterion, across all three SPL, that was perceived as being quick was architecture. In addition, PLC viewed process walkthrough as a fast method. One main difference between the three SPL was that PLA and PLB participants viewed electronic chat as one of the fastest criterion, while PLC participants preferred forums.

**Table 5.** Relative importance of speed

	Speed					
	Local			Global		
	PLA	PLB	PLC	PLA	PLB	PLC
<b>Technical communication mechanisms</b>						
Architecture	7.8%	9.5%	8.4%	14.2%	12.1%	12.3%
Change Management	7.4%	4.4%	7.5%	8.1%	5.2%	7.5%
Code Walkthrough	9.4%	10.2%	11.6%	7.9%	6.5%	6.1%
Process Walkthrough	8.3%	13.3%	9.0%	7.1%	8.9%	11.7%
Progress Report	7.2%	4.6%	6.4%	8.2%	5.1%	8.2%
<b>Local peer-to-peer communication mechanisms</b>						
Face-To-Face	<b>17.1%</b>	<b>20.6%</b>	<b>16.7%</b>	10.8%	9.0%	8.6%
Regular Meeting	11.0%	8.5%	10.3%	9.5%	11.8%	11.2%
Visiting Engineer	11.3%	6.9%	9.0%	12.9%	<b>19.6%</b>	<b>14.4%</b>
<b>Long distance peer-to-peer communication mechanisms</b>						
Forums	8.4%	4.9%	8.3%	5.8%	4.2%	14.0%
Electronic Chat	12.2%	17.1%	12.9%	<b>15.5%</b>	17.5%	6.0%

### 5.3 Communication Rich Information (RQ1.2)

In terms of providing rich information for local site communication, a mix of *technical* (architecture) and *local peer-to-peer* (face-to-face) communication mechanisms were perceived to provide the richest information (see Table 6). For PLA and PLC, face-to-face provided the richest information, while PLB participants viewed architecture as the richest source. It is interesting to note that PLB participants viewed process walkthrough as a criterion that provided rich information, while PLA and PLC participants perceived process walkthrough among the least effective communication mechanism for providing rich information. For global sites communication, all three SPL agreed that architecture provided the richest information. Moreover, a visiting

engineer was viewed, by all three SPL, to be a good communication method for rich information. The main difference between the three SPL related to how long distance peer-to-peer provided rich information. PLA and PLB participants perceived electronic chat as a good source of rich information, while PLC participants viewed forums as a good source.

**Table 6.** Relative importance of richness

	Richness					
	Local			Global		
	PLA	PLB	PLC	PLA	PLB	PLC
<b>Technical communication mechanisms</b>						
Architecture	12.6%	<b>18.2%</b>	14.4%	<b>17.7%</b>	<b>18.1%</b>	<b>17.4%</b>
Change Management	6.7%	7.9%	7.0%	7.6%	5.3%	6.2%
Code Walkthrough	11.6%	9.7%	8.2%	10.0%	8.5%	8.8%
Process Walkthrough	6.7%	14.1%	7.3%	6.5%	8.6%	11.3%
Progress Report	6.4%	4.4%	5.6%	8.7%	6.1%	7.5%
<b>Local peer-to-peer communication mechanisms</b>						
Face-To-Face	<b>17.7%</b>	17.5%	<b>16.3%</b>	9.7%	7.9%	6.2%
Regular Meeting	9.7%	5.7%	10.3%	6.9%	8.2%	9.7%
Visiting Engineer	10.0%	6.5%	9.0%	15.5%	14.6%	13.6%
<b>Long distance peer-to-peer communication mechanisms</b>						
Forums	8.0%	6.1%	12.1%	5.7%	5.6%	13.3%
Electronic Chat	10.5%	9.9%	9.9%	11.8%	17.1%	6.0%

#### 5.4 Communicate Large Volume of Information (RQ1.3)

Looking at which communication mechanism provides the largest volume of information, both PLA and PLB participants viewed *technical* communication mechanisms as providing the largest volume of information. Architecture was perceived as the preferred criterion when sharing large volumes of information. PLA participants viewed code walkthrough and PLB participants viewed process walkthrough as the second most suitable criterion respectively. Unlike PLA and PLB participants, PLC participants viewed forums (*long distance peer-to-peer*) as the criterion that provided the largest volume of information, which is a surprising finding. Moreover, regular meetings were perceived as equally good as architecture for distributing large volumes of information. For global sites communication, the results with regards to providing large volume of information were similar to the results of providing rich information. Architecture was viewed by all three SPL as the preferred criterion for sharing large volumes of information. Moreover, PLA and PLB participants perceived electronic chat as a way of sharing large volumes, while PLC preferred forums. In addition, PLA and PLB participants identified visiting engineers, while PLC participants viewed regular meetings, as other good communication methods for sharing large volumes of information.

## 6 Discussion

The result of the survey shows that a number of communication mechanisms are more important than others in local and global environments. While no single mechanism meets all three needs of speed, richness and volume perfectly, when examining the results across the three product lines, the prevalence of architecture in providing rich information in large volumes, for both local and global communication, can be clearly observed. This has important implications for SPL engineering, which uses product line architecture as a driving force in developing software products. It indicates that traditional SPL engineering practices and artifacts have the ability to act as reusable items, reused as a communication mechanism enabler. This finding is expected as architecture establishes a method of effective communication through a common vocabulary [1]. Not only does architecture make large-scale reuse possible, by establishing component definitions and proper interfaces, it also provides for orientation of different development teams systematically producing different parts of the system [15], while the impact of distance, language, and culture are minimized [16]. This provides a potential solution for a fundamental challenge for GSE: the communication and coordination between distributed teams working on different areas of product development. This also implies that SPL engineering can be used efficiently and effectively in organizations that have globally distributed teams.

Further examination of the results by product lines show that, communication between development teams, locally or globally, is also dependent on their development practices. Taking the communication practices for local sites into consideration, it was mostly peer-to-peer communication methods, such as face-to-face conversations and electronic chat that provided information in a relatively fast manner across all product lines. Communication mechanisms that allow for mass distribution of information, such as forums and architecture, generally rated lower in terms of speed of distribution. This may be attributed to the fact that such mechanisms were generally qualitative and textual in nature, requiring time to comprehend. In the case of global communication, face-to-face conversations were replaced by an equivalently fast communication mechanism, a visiting engineer. This indicates that personal contact is largely associated with faster dissemination of information. However, the importance of a visiting engineer is not consistent with the findings of Smite, which found that communication conducted through meeting in person was very seldom used [24]. In addition, McDonough et al. found that email and databases were the fastest communication tool, while face-to-face was considered as providing the richest information [15].

The importance of local peer-to-peer communication mechanisms is not surprising; however, having regular meetings (part of local peer-to-peer communication mechanisms) was not viewed as being among the top three fastest communication mechanisms. This indicates that the participants referred to informal and ad-hoc communication when talking about face-to-face communication. This result is consistent with several studies [4, 6, 10, 12] that point out the importance of informal communication for the success of GSE. One reason why regular meeting was not viewed in the top three fast communication mechanisms may be the difficulties of using formal communication channels to handle unexpected events [10].

The influence of development practices on the communication mechanisms used was more evident in the criteria of richness and volume. In particular, electronic

forums were rated considerably lower in PLA due to developers working more closely on their own products, rather than relying on shared artifacts. Greater use of shared artifacts correlates with an increase in the use of mechanisms that provide large volumes of rich information. This was evident in PLC, which heavily utilizes shared artifacts, where electronic forums and architecture were ranked higher when compared to other mechanisms.

In summation, the prevalence of architecture in providing rich information in large volumes, both for local and global communication can be observed. This may indicate that software architecture can enable communication in SPL in a globally distributed software development context.

## 7 Threats to Validity

In this section, threats to validity are discussed. We consider the four perspectives of validity and threats [28], i.e. construct, conclusion, internal and external validity.

*Construct validity* is concerned with the relation between theories behind the research and the observations. The variables in our research are measured through interviews, including open-ended aspects where the participants are asked to express their own opinions, and a questionnaire. To avoid evaluation apprehension [28], complete anonymity from other participants, and researchers was guaranteed. Another validity threat lies in the questionnaire that asked the participants to rank communication practices and include additional practices if the list provided to them was inadequate. Participants may have thought that it was easier to rank the provided factors than propose new factors. Hence, important communication practices may be missing.

Threats to *conclusion validity* arise from the ability to draw accurate conclusions. The interviews were conducted at different branches and each interview was done in one work session. Thus, answers were not influenced by internal discussions. To obtain highly reliable measures and to avoid poor question wording and poor layout, several pilot studies were conducted.

*Internal validity* is related to issues that affect the causal relationship between treatment and outcome. Threats to internal validity include instrumentation, maturation and selection threats. In our study, the research instruments were developed with close reference to literature relating to GSE, and influenced by a previously administered and validated research instrument [14], which mitigates instrumentation threat. Maturation threats are handled by keeping the interview session to 60 minutes.

*External validity* is concerned with the ability to generalize findings beyond the actual study. Qualitative studies rarely attempt to generalize beyond the actual setting since it is more concerned with explaining and understanding the phenomena. The nature of qualitative designs also makes it impossible to replicate since identical circumstances cannot be recreated. However, understanding the phenomena may help in understanding other cases and situations. The participants selected may not adequately reflect the diversity of opinion on current practice communication mechanisms. The small sample size used in this study may also indicate that conclusions made may not be generalized across the software industry. Hence, the results of the study must be interpreted with caution when moving away from the characteristics of the studied case organization.

## 8 Conclusion

This paper presents the results of an empirical study that examines the importance of commonly used communication mechanisms across local and global development sites. To the best of our knowledge, there is no empirical study that specifically examines the aspects of communication mechanism, in product lines, for their ability to transmit information quickly, to transmit rich information and to transmit large volumes of information.

This study has shown that some communication mechanisms are more important than others in a local and global environment. While there are some differences between the three product lines, peer-to-peer communication mechanisms are perceived to be particularly important at a local level, and to provide a faster speed of communication at a global level. Software architecture was generally perceived to communicate large volumes of rich information at both a local and global level. Participants across all three product lines understood the relative importance of architecture in the global environment when compared to other communication mechanisms. This indicates that SPL engineering has the ability to utilize a globally distributed development environment and that a heavy reliance on software architecture can enable communication in SPL in a globally distributed software development context.

This study is only a first step in understanding commonly used communication mechanisms in SPL in a global development context. Given the limitations with the sample size, there is opportunity for future research to replicate this study across various cases and across different industries. Future work should focus on the detailed utilization of software architecture or other SPL engineering assets for GSE. This could provide improvements for these artifacts to be more useful in a global context. It also could provide ideas for improving the global SPL engineering process in terms of work distribution and management.

## References

1. Bass, L., Clements, P., Kazman, R.: *Software Architecture in Practice*. Addison-Wesley Professional (2003)
2. Berenbach, B.: An Introduction to Global Product Line Requirements Engineering. In: *Proc. of the Second Int. Conference on Global Software Engineering*, pp. 300–301 (August 2007)
3. Bosch, J., Bosch-Sijtsema, P.: From integration to composition: On the impact of software product lines, global development and ecosystems. *Journal of Systems and Software* 831, 67–76 (2010)
4. Carmel, E., Agarwal, R.: Tactical approaches for alleviating distance in global software development. *IEEE Software* 18, 22–29 (2001)
5. Conchuir, E.O., Ågerfalk, P.J., Olsson, H.H., Fitzgerald, B.: Global Software Development: Where are the benefits? *Communications of the ACM* 52, 127–131 (2009)
6. Deridder D.: A concept-oriented approach to support software maintenance and reuse activities. In: *Workshop on Knowledge-Based Object-Oriented Software Engineering* (2002)
7. Falbo, R.A., Menezes, C.S., Rocha, A.R.: Using ontologies to improve knowledge integration in software engineering environments. In: *Proc. of the 4th Conference on Information Systems Analysis and Synthesis* (1998)
8. Fricker, S., Gorschek, T., Glinz, M.: Goal-Oriented Requirements Communication in New Product Development. In: *Proc., Second Int. Workshop on Software Product Management* (2008)

9. Griffin, A., Hauser, J.: Integrating R&D and Marketing: A Review and Analysis of the Literature. *Journal of Product Innovation Management* 13, 191–215 (1996)
10. Herbsleb, J.D., Grinter, R.E.: Architectures, Coordination, and Distance: Conway's Law and Beyond. *IEEE Software* 16, 63–70 (1999)
11. Herbsleb, J.D., Mockus, A.: An Empirical Study of Speed and Communication in Globally Distributed Software Development. *IEEE Transactions on Software Engineering* 29, 481–494 (2003)
12. Herbsleb, J.D., Mockus, A., Finholt, T.A., Grinter, R.E.: An Empirical Study of Global Software Development: Distance and Speed. In: *Proc. 3rd International Conference on Software Engineering*. Inst. of Elec. and Elec. Eng, pp. 81–90 (May 2001)
13. Kraut, R.E., Streeter, L.A.: Coordination in software development. *Communications of the ACM* 38, 69–81 (1995)
14. McDonough, E.F., Kahn, K.B., Griffin, A.: Managing Communication in Global Product Development Teams. *IEEE Transaction on Engineering Management* 46, 375–386 (1999)
15. Niinimäki, T., Lassenius, C.: Experiences of Instant Messaging in Global Software Development Projects: A Multiple Case Study. In: *Proc. IEEE International Conference on Global Software Engineering*, pp. 55–64. IEEE (August 2008)
16. Ovaska, P., Rossi, M., Marttiin, P.: Architecture as a coordination tool in multi-site software development. *Software Process: Improvement and Practice* 8, 233–247 (2004)
17. Paech, B., Dörr, J., Koehler, M.: Improving Requirements Engineering Communication in Multiproject Environments. *IEEE Software* 22, 40–47 (2005)
18. Perry, D.E., Staudenmayer, N.A., Votta, L.G.: People, organizations and process improvement. *IEEE Software* 11, 36–45 (1994)
19. Regnell, B., Brinkkemper, S.: Market-Driven Requirements Engineering for Software Products. In: Aurum, A., Wohlin, C. (eds.) *Engineering and Managing Software Requirements*, pp. 287–308. Springer (2005)
20. Robson, C.: *Real World Research*. Blackwell, Oxford (2002)
21. Sharma D.: *Blueprint of Success: Creating Software Product Value through Product Line Engineering*, Honours Thesis, School of Information Systems, Technology and Management, University of New South Wales, Australia (2007)
22. Sharma, D., Aurum, A., Paech, B.: Business Value through Product Line Engineering – An Empirical Study. In: *Proc., 34th Euromicro Conference on Software Engineering and Advanced Applications*, pp. 167–174 (September 2008)
23. Silverman, D.: *Interpreting Qualitative Data*. Sage Publication, London (2001)
24. Šmite, D.: Global Software Development Projects in One of the Biggest Companies in Latvia: Is Geographical Distribution a Problem? *Software Process Improvement and Practice* 11, 61–76 (2006)
25. Šmite, D., Wohlin, C., Gorschek, T., Feldt, R.: Empirical Evidence in Global Software Engineering: A Systematic Review. *Empirical Software Engineering* 15, 91–118 (2010)
26. Ullah, M.I., Ruhe, G.: Towards Comprehensive Release Planning for Software Product Lines. In: *Proc. of the 1st International Workshop on Software Product Management (ISPM 2006)*. Inst. of Elec. and Eng. Computer Society, pp. 51–55 (September 2006)
27. Van De Weerd, I., Brinkkemper, S., Nieuwenhuis, R., Versendaal J., Bijlsma, L.: *A Reference Framework for Software Product Management*, Technical Report UU-CS-2006-014, Department of Information and Computing Sciences Utrecht University (2006)
28. Wohlin, C., Runeson, P., Höst, M., Ohlson, C., Regnell, B., Wesslén, A.: *Experimentation in Software Engineering: An Introduction*. Kluwer Academic, Boston (2000)