Evaluating the Usefulness and Ease of Use of a Groupware Tool for the Software Architecture Evaluation Process

Muhamad Ali Babar¹  Dietmar Winkler²  Stefan Biffl²
¹Lero, University of Limerick, Ireland ²TU Wien, Institute of Software Technology, Vienna, Austria
¹malibaba@lero.ie, ²{winkler, biffl}@qse ifs.tuwien.ac.at

Abstract
We have developed a framework for groupware tool support for the software architecture evaluation process in the context of global software development. We have empirically assessed the effectiveness of the groupware-supported software architecture evaluation process in a set of controlled experiments. While we found that groupware-supported distributed meetings can be very effective, we saw the need to investigate users’ acceptance of the tool used in these empirical studies. In this paper we report on the “perceived usefulness” and “ease of use” of the groupware tool based on the adapted Davis’ Technology Acceptance Model (TAM), a widely used general-purpose instrument for measuring users’ attitude towards a particular technology. Main results from analyzing the TAM data are: a majority of the participants found the tool quite useful and easy to use for supporting collaborative tasks like architecture evaluation; a majority of the respondents was also very positive about the regular use of the tool for collaborative tasks in the future. However, there was considerably less support for preferring a distributed tool-based meeting to a face-to-face meeting.

Keywords: Software architecture evaluation, groupware support, empirical study, Technology Acceptance Model, global software development.

1. Introduction
It is widely recognized that software architecture evaluation is an effective quality assurance technique that helps identify potential architectural risks and questionable design decisions [1][2]. Most of the well-known software architecture evaluation methods are scenario based [3], such as the Architecture Tradeoff Analysis Method (ATAM) [1], Performance Assessment of Software Architecture (PASA) [4], and Architecture-Level Maintainability Analysis (ALMA) [5]. These methods rely extensively on the collaborative efforts of multiple stakeholders working in face-to-face meetings [3].

However, co-locating a large number of stakeholders is an expensive and time-consuming exercise, which can create logistical problems such as scheduling difficulties and project delays [6]. These problems typically increase, if stakeholders are geographically separated and have to travel, which is highly likely as distributed software development is increasingly becoming the norm than a fad [7][8]. Such issues may hinder the widespread adoption of disciplined architecture evaluation practices. Our research challenge was to find and assess an effective and efficient way of enabling physically dispersed stakeholders to participate in the software architecture process without having to travel.

Based on the positive reports on the use of groupware systems to support several software engineering activities (such as requirements negotiation, software design, and software inspection [9]-[13]), we have developed a framework for using a groupware system for the software architecture evaluation process [14]. We conducted empirical studies to assess the effectiveness of one of the activities (i.e., developing scenarios for characterizing required quality attributes) and found that developing scenarios can be supported by a groupware system without compromising the quality of the outcome [14][15].

Having proposed a groupware-supported software architecture evaluation process, a key research issue was to identify the features required of a groupware system to support the proposed process and gain a better understanding of how groupware support facilitates or hinders social processes involved in the proposed (distributed) process. In this respect, we were also interested in assessing a generic groupware tool, LiveNet, used in our experiments. A good understanding of our study’s participants’ attitude about LiveNet is expected to help us to decide whether or not and how LiveNet should be tailored to support the proposed process and determine the features required of a groupware system [14].

When measuring people’s attitude about a particular technology, a researcher needs to rely on subjective meas-
...ures for inferring conclusions as no objective measure can help in deciding whether a particular technology is “good” or “bad” from users’ perspective. For this research, we decided to adapt and apply the well-known Technology Acceptance Model (TAM) [17]. TAM postulates that two particular user beliefs, “usefulness” and “ease of use”, are the basic determinants of technology acceptance behavior of a user [18]. We adapted the TAM questionnaire for assessing LiveNet. We modified the wordings of the standard 12 questions of TAM and 2 questions on self-predicted future use of a technology introduced by Laitenberger and Dreyer in their study of evaluating a tool using TAM [18]. Furthermore, we included 3 open-ended questions to identify the architecture evaluation tasks that seemed to be well- or poorly-supported by LiveNet and to determine the requirements for a groupware tool for supporting the proposed process [16]. This paper reports the results from an analysis of the data collected using the adapted TAM.

The structure of the remainder of the paper is as follows. Section 2 summarizes related work on software architecture evaluation, groupware support, and technology acceptance measurement. Section 3 lists the research issues and presents the measurement tool; Section 4 briefly describes the empirical study objectives and logistics; Section 5 provides the results of the data analysis and their interpretation. Section 6 discusses the results and Section 7 summarizes conclusions and lessons learned for future work.

2. Background and Motivation

This section summarizes related work on the software architecture evaluation process, collaboration tasks, tool support for collaboration, and measurement of technology acceptance.

2.1 Software architecture evaluation process

Several methods (such as ATAM [1] and ALMA [5]) have been proposed to support architecture evaluation. Most of these methods are structurally similar but there are a number of differences among their activities and techniques [3]. One of the commonalities among these approaches is meeting-based activities. The requirement to hold meetings is partially created by the very nature of these approaches. For example, “Quality Attribute Workshops (QAWs) [1]” of architecture evaluation process are intended to elicit the quality goals of a system by generating scenarios. Stakeholders also prioritize the generated scenarios according to business goals [2]. Architecture evaluation workshops also provide an opportunity for stakeholders to become familiar with the proposed architectural approaches, ask questions, address issues that appear to be risk prone. All these activities require group discussions and decision making activities, which necessitate face-to-face meetings according to the current architecture evaluation methods. However, co-locating stakeholders can pose several logistical and organizational challenges, especially in the context of global software development.

2.2 Groupware for software development

Face-to-face meetings for software inspections are likely to incur substantial cost and lengthen the development process [8]. Some studies even have called into question the value of face-to-face inspection meetings [19]. Groupware-supported inspection processes have been successfully evaluated as a promising way to minimize meeting costs, maximize asynchronous work, and conserve precious organizational resources [13]. The Requirements Engineering (RE) community has also successfully used groupware to enable distributed teams of stakeholders to perform different RE tasks. For example, Liou and Chen integrated joint application development (JAD) and groupware to support requirements acquisition and specification activities [20]. Damian and her colleagues reported on successful experiments with using a web-based collaborative tool to support requirements negotiation meetings [10]. Boehm and his colleagues developed a groupware tool to support their EasyWinWin requirements negotiation methodology [9] and integrated a case tool to improve the support for requirements engineering tasks [21].

2.3 Groupware for architecture evaluation

Encouraged by the successful experiences of using groupware technologies to enable geographically distributed teams in performing various software engineering activities, we proposed a groupware-supported distributed software architecture evaluation process. The proposed process is expected to address a number of logistical issues that characterize the current software architecture evaluation approaches. According to this framework, a number of activities (such as evaluation planning, scenario gathering, scenario prioritization, and scenario mapping) can successfully be performed in a distributed environment using a groupware system [22].

Having developed the groupware-supported distributed software architecture evaluation process, our next step was to assess the effectiveness of the proposed process and determine the requirements for groupware system to support the proposed process. We also analyzed the tools used for research on groupware support for software development activities. We found that a distributed software architecture evaluation process has several unique aspects, which cannot be supported with the tools developed or tailored for other software development activities (see [16] for details results of our analysis).

That was why for our experiments, we decided to use a generic collaborative application, LiveNet, based on its promising features and ease of availability for research.

431
purposes. LiveNet system has been developed by the Collaborative Systems Laboratory at the University of Technology, Sydney. LiveNet provides a generic workflow engine and different features to support collaboration among geographically distributed members of a team. LiveNet enables users to create workspaces and define elements of a particular workspace such as activities, tasks, roles, participants, and communication tools [23]. A generic groupware system like LiveNet typically provides functionality for distributed collaboration tasks such as brainstorming, document sharing, discussion forum for consensus building, notification mechanism, and asynchronous and synchronous communication channels.

2.4 Technology Acceptance Measurement

The Technology Acceptance Model (TAM) [17] aims at assessing user beliefs about the usefulness and ease of use of a technology that is expected to support their work. Perceived usefulness is defined as “the degree to which a person believes that using a particular technology would enhance his or her job performance”, while perceived ease of use refers to “the degree to which a person believes that using a particular system would be free of effort.” [17] According to the theory of reasoned action [24], these factors are strongly correlated to the intention of actually using the technological innovation, if it is available, i.e., self-reported future use (see Figure 1).

TAM has been widely applied in technology assessment with reliable results when users have worked with the technology over time. King and He [25] report on the results of a meta-analysis of 88 published TAM studies supporting the validity and robustness of the instrument over a wide range of applications. Satzinger and Olfman [26] report the results of two surveys on the perceived usefulness of group support for collaborative tasks using TAM model. They found that their subjects clearly preferred groupware support for work between meetings rather than for the support of distributed or face-to-face meetings. Laitenberger and Dreyer [18] provide a good introduction to TAM adaptation for software tool acceptance measurement for software inspection data entry.

3. Research Objective

The main goal of this study is to evaluate a groupware tool for supporting the software architecture process and understand the requirements for tailoring and extending that tool. This research goal can be broken down into the following research issues:

1. **Usefulness and ease of use**: Survey whether the subjects find LiveNet’s support for collaborative tasks of architecture evaluation useful and easy to use; and measure the reliability of the sets of questions for usefulness and ease of use.

2. **Self-predicted future use**: Survey whether the subjects report willingness to use the groupware tool in their future work.

3. **Preference of meeting style**: Survey whether the subjects report a preference for tool-supported distributed meetings or for face-to-face meetings for scenario generation (see Section 2.1). This issue looks at the preference for a process variant in combination with a tool support.

4. **Tasks supported well or poorly by LiveNet**: Obtain participants’ opinion about the architecture evaluation tasks that are supported well or poorly by LiveNet.

5. **Subject experience and TAM responses**: Analyze whether subjects with low, medium, or high experience in potentially relevant collaborative work aspects demonstrate different attitude towards groupware tool’s, LiveNet, support for software architecture evaluation process. Factors of participant experience, such as prior use of collaborative tools, may have significant impact on their perceived usefulness, ease of use of LiveNet, and preference for tool-supported distributed or face-to-face meetings.

4. The Adapted TAM Measurement Tool

We found TAM a suitable starting point to develop an adapted measurement tool for assessing the “ease of use” and “usefulness” of LiveNet to support the collaborative tasks in the context of software architecture evaluation. Figure 1 shows our TAM-based measurement model adapted from [17][18]. For each of the three main constructs “perceived usefulness” (U), “ease of use” (E), and “self-predicted future use” (S) there are sets of questions that measure the construct. Furthermore, a set of questions measures prior experience of subjects that may influence TAM results.

![Figure 1: Relationships of questionnaire parts to higher-level TAM constructs.](image-url)
We adapted the TAM as follows (see Table 1):

- The tool object of the questions in the questionnaire was named "collaborative tool like LiveNet"
- The process investigated in the questionnaire was named "collaborative tasks" with a focus on software architecture evaluation process.
- We used a Likert scale that measured perceived usefulness with a six-point semantic differential scale which asked for the degree of likelihood that a statement about the usefulness of an application is true ("extremely likely" ... 1 to "extremely unlikely" ... 6) instead of seven-point scales used in [17][18], so subjects had to express their tendency towards a positive or negative evaluation more clearly as there was no neutral value available.

- Furthermore, we added open-ended questions to identify architecture evaluation tasks perceived by the respondents to be supported well or poorly by a collaborative tool like LiveNet.


The empirical study to assess LiveNet was conducted as part of a controlled experiment assessing the groupware-supported software architecture evaluation process. This experiment focused on comparing the effectiveness of developing scenario profiles for architecture evaluation in face-to-face and groupware-supported meeting arrangements. [15]. This section provides a summary of the controlled experiment relevant to this paper: an empirical assessment of a collaborative tool, LiveNet, for supporting different activities of the software architecture evaluation process (i.e., scenario development for this study), discusses the study's limitations, and subject experience prior to forming subject groups for data analysis.

5.1 Empirical study description

There were two objectives for using TAM: a) assessing LiveNet’s usefulness and ease of use for supporting different activities of the distributed software architecture evaluation process and b) identify the requirements for a groupware system to support the proposed process. A set of requirements for a groupware identified by the participants of our empirical studies has been reported in [16]. The questionnaire also gathered demographic data.

Before the experiment, subjects answered questionnaires on their background and experience. In the class period before the experiment students had already used LiveNet on a regular basis to get familiar with its basic functions, such as setting up a collaborative group, exchanging documents, and conducting discussions in synchronous (chat) and asynchronous settings (using a forum). In the experiment the participants developed quality sensitive scenarios for evaluating software architecture.

The experiment design provided all the participants with an opportunity to perform the required task once in a co-located arrangement and once in a distributed arrangement using the software requirements of two different systems. We used an AB/BA cross-over experiment design, which required the participants to develop scenarios twice, once in a traditional face-to-face setting and once in a distributed setting using LiveNet [15]. Further details on the experiment design, logistics, and results can be found in [15].

5.2 Tool training and usage

The study participants were supposed to use a groupware system during the scenario development activity. Hence, the participants received training on using a col-
laborative tasks, LiveNet, for performing various collabora
tive tasks. Furthermore, this collaborative application
was an integral part of the course as the students were
required to use the application to participate in the dis-
cussions on course related topics. This requirement was
expected to enable the participants to become familiar
with LiveNet’s basic functions, such as setting up a col-
laborative group, exchanging documents, and conducting
discussions in synchronous (chat) and asynchronous set-
tings (using a forum). Thus, when the respondents an-
swered the TAM-based assessment instrument, they had
already used LiveNet for more than six weeks not only to
perform the experimental tasks but also to perform
group-based tutorial tasks and assignments.

5.3 Subjects and their experience

In addition to the TAM-based questionnaire to obtain
self-reported information to assess the suitability of Li-
veNet to support distributed software architecture evalu-
tion, the questionnaire collected demographic data such
as experience level, gender, age etc. The respondents to
the questionnaire were 3rd- and 4th-year students of soft-
ware engineering and computer engineering degrees en-
rolled in a course on Total Quality Management offered
by the University of New South Wales (UNSW), Aus-
tralia. We decided to include only 113 responses in our final
analysis as many of the participants of our empirical
study did not provide their identification number for data
validation purposes.

Table 2a. Classification of participant experience.

<table>
<thead>
<tr>
<th>Experience Category</th>
<th>LOW</th>
<th>MEDIUM</th>
<th>HIGH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Team work experience (E_TEAM)</td>
<td>&lt;1 year</td>
<td>1-3 years</td>
<td>&gt;3 years</td>
</tr>
<tr>
<td>Collaborative work experience (E_COLL)</td>
<td>&lt;3 months</td>
<td>3-12 months</td>
<td>&gt;12 months</td>
</tr>
<tr>
<td>Experience in Working with collaborative Tools (E_TOOL)</td>
<td>&lt;3 months</td>
<td>3-18 months</td>
<td>&gt;18 months</td>
</tr>
<tr>
<td>Experience with Analysis and Design (E_AD)</td>
<td>&lt;3 months</td>
<td>3-18 months</td>
<td>&gt;18 months</td>
</tr>
</tbody>
</table>

The participants had a strong technical background,
and varying degrees of work experience. For conducting
collaborative tasks and appreciating tool support, we
identified four variables in the demographic data and
assigned them to a three-level classification schema (low,
medium and highly experienced participants). Table 2a
describes the four most important variables and the map-
ing of classified experience levels. Table 2b summarizes
the distributions of participants regarding the selected
experience variables. Note that we conducted data analy-
sis with an overall number of 113 participants. Overall,
the participants reported medium to high experience in
team work, but mostly low experience with collaborative
work and also collaborative tools.

5.4 Validity threats

For the controlled experiment on the effectiveness of
collaborative tasks in co-located and distributed settings,
we identified and addressed a number of internal and ex-
ternal threats. We have provided a detailed description of
these threats and measures taken to counter them in [15].
Here, we list external validity threats that are considered
to be relevant to the tool evaluation in this paper.

Student subjects. The participants were undergraduate
students with predominantly technical background, who
may find a collaborative tool like LiveNet more easy to
use than general users [25]. But this aspect is an unavoid-
able factor when conducting an experiment with student
participants.

Process validity. Another threat may be task realism if
the scenario development process used in our experiment is
not representative of industrial practices. The participants
of our study followed a scenario development process that
is quite similar to the one used for most of the scenario-
based software architecture evaluation methods such as
ATAM [1]. The collaborative processes were documented
in the study to ensure that participants actually followed
the prescribed processes.

Tool familiarity. The study participants were supposed
to use a groupware system during the scenario develop-
ment activity. Hence, the participants received training on
using a collaborative tool, LiveNet, for performing various
collaborative tasks. Furthermore, this collaborative appli-
cation was an integral part of the course as the students
were required to use the application to participate in the
discussions on course-related topics. Thus, when the re-
pondents answered the TAM-based assessment instrument,
they had already used LiveNet for more than six weeks not only to perform the experimental tasks but also to
perform group-based tutorial tasks and assignments.

6. Data Analysis and Results

This section presents the results from data analysis for
the research issues mentioned in Section 3.

6.1 Usefulness and Ease of Use

Usefulness and Ease of Use are important measures for
a tool acceptance. We used TAM-based questionnaire to
obtain respondents’ opinion about LiveNet. Table 3 shows
the descriptive statistics for the TAM’s questions U_1 to U_6 and E_1 to E_6. The numerical results correspond to a Likert scale from 1 (extremely likely) to 6 (extremely unlikely) (see detailed questionnaire in Section 3). An average response between 2 (“quite likely”) and 3 (“slightly likely”) seems overall a cautiously positive result, where many subjects responded very positively, however, a significant number of the subjects was not convinced about the usefulness and ease of use of LiveNet’s support for collaborative tasks involved in software architecture evaluation process.

Table 3: Mean and Std.Dev. for adapted TAM constructs (Likert scale: 1 to 6).

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>Std. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Work more Quickly (U1)</td>
<td>2.98</td>
<td>1.118</td>
</tr>
<tr>
<td>Improve Performance (U2)</td>
<td>2.85</td>
<td>0.966</td>
</tr>
<tr>
<td>Increase Productivity (U3)</td>
<td>3.01</td>
<td>1.048</td>
</tr>
<tr>
<td>Effectiveness (U4)</td>
<td>2.81</td>
<td>1.023</td>
</tr>
<tr>
<td>Makes Job Easier (U5)</td>
<td>2.77</td>
<td>1.086</td>
</tr>
<tr>
<td>Useful (U6)</td>
<td>2.78</td>
<td>1.041</td>
</tr>
<tr>
<td>USEFULNESS</td>
<td>17.20</td>
<td>4.945</td>
</tr>
<tr>
<td>Easy to Learn (E1)</td>
<td>2.28</td>
<td>.881</td>
</tr>
<tr>
<td>Easy to Perform (E2)</td>
<td>2.80</td>
<td>.888</td>
</tr>
<tr>
<td>Clear and Understandable (E3)</td>
<td>2.63</td>
<td>.928</td>
</tr>
<tr>
<td>Easy to become Skilful (E4)</td>
<td>2.10</td>
<td>.756</td>
</tr>
<tr>
<td>Easy to Remember (E5)</td>
<td>2.23</td>
<td>.813</td>
</tr>
<tr>
<td>Easy to Use (E6)</td>
<td>2.29</td>
<td>.883</td>
</tr>
<tr>
<td>EASE OF USE</td>
<td>14.50</td>
<td>3.951</td>
</tr>
</tbody>
</table>

One reason for such response might be the low level of experience and little exposure to collaborative tool among the respondents (see section 4.3 for details). Note that about 56% of the participants recorded very little prior experience in collaborative work, and about 60% reported very little prior experience in using collaborative tools.

6.2 Self-predicted future use

Table 4a reports self-predicted future (S1) use of a collaborative tool like LiveNet if it is available. It is an intention of using a technological innovation, provided it is available. It appears that on average the participants intended to use the tool regularly if it is available.

Table 4a: Mean and Std.Dev. for perceived usefulness and preferred tool support.

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>Std. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Usefulness (S1)</td>
<td>2.73</td>
<td>0.982</td>
</tr>
<tr>
<td>Prefer Tool-Supported Distributed Meeting to Face-to-Face Meeting (S2)</td>
<td>3.98</td>
<td>1.369</td>
</tr>
</tbody>
</table>

6.3 Preference of meeting style

Table 4a also shows that the participants seemed to prefer face-to-face meetings to tool-supported distributed meetings for scenarios development for software architecture evaluation. Table 4b summarizes the type of meeting preferred by the participants for generating scenarios.

Table 4b: Type of Meeting Preferred.

<table>
<thead>
<tr>
<th></th>
<th>Number</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Face-to-face arrangement</td>
<td>93</td>
<td>82%</td>
</tr>
<tr>
<td>Distributed arrangement using a collaborative tool</td>
<td>20</td>
<td>18%</td>
</tr>
</tbody>
</table>

Most participants preferred a face-to-face meeting to a tool-supported distributed arrangement. This question covers two different aspects: (a) distributed arrangement and (b) usage of a collaborative tool. Most of the participants might see co-located architecture evaluation easier and more efficient than distributed arrangements, possibly due their higher experience with a co-located work style which allows them to conduct co-located tasks more efficiently (but not necessarily more effectively). Moreover, These results are also consistent with the findings of our two controlled experiments in which majority of the participants preferred face-to-face meetings to groupware-supported meetings despite the quality of the output of the tool-supported process was better than the output of the face-to-face meetings [14][15].

6.4 Tasks well and poorly supported

There were two open-ended questions aimed at obtaining the participants’ opinion about the software architecture evaluation tasks that they thought either well or poorly supported by LiveNet. We used an emergent coding scheme to code the responses to the open-ended questions. For encoding the data for both questions, two researchers independently reviewed the data and built two coding checklists based on the identified tasks. They resolved all the issues related to the reliability of the checklist before forming a final checklist, which was applied to the responses to annotate the data with the task themes. Finally, frequency for each task identified (well or poorly supported) was taken.

Table 5: Tasks identified as supported well or poorly by LiveNet.

<table>
<thead>
<tr>
<th>Tasks supported well</th>
<th>Frequency</th>
<th>Tasks supported poorly</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brainstorming</td>
<td>79</td>
<td>Project management</td>
<td>15</td>
</tr>
<tr>
<td>Documents sharing</td>
<td>79</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Synchronous communication</td>
<td>57</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Asynchronous communication</td>
<td>24</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 6a: Cronbach’s Alpha for adapted TAM constructs.

<table>
<thead>
<tr>
<th>Cronbach’s alpha</th>
</tr>
</thead>
<tbody>
<tr>
<td>Usefulness</td>
</tr>
<tr>
<td>Ease of Use</td>
</tr>
</tbody>
</table>

6.5 Subject experience and TAM responses

Factors of participant experience, such as prior use of collaborative tools, may have a significant impact on their perceived usefulness, ease of use of LiveNet’s support for architecture evaluation, and preference for tool supported distributed or face-to-face meetings. Hence, we analyzed the data to find out whether subjects with low, medium, or high experience in collaborative work have different attitudes towards LiveNet’s support for architecture evaluation. For this purpose, we applied the non-parametric Mann-Whitney Test at an alpha level of 0.05. Data analysis revealed only little impact of the experience factors on the participants’ attitude. For the interested reader we have reported the detailed results about the subject experience and TAM response in [27].

6.6 TAM reliability and factor validity

In this section we report on the results from analysis of the data for investigating the relationships between detailed sets of questions in the TAM parts.

*TAM reliability evaluation* - reliability can be seen as the degree of accuracy of measures within an empirical study. A Cronbach’s Alpha reliability level that exceeds a threshold level of 0.8 indicates a reliable measure [18][28]. Thus, both aspects of the adapted TAM can be considered reliable (see Table 6a).

| Table 6a: Cronbach’s Alpha for adapted TAM constructs. |
|------------------|------------------|
| Usefulness        | 0.88 |
| Ease of Use       | 0.81 |

*TAM factor validity evaluation* - factor analysis calculates a few factors that are derived in order to represent well the variation of many data dimensions, in our case questionnaire questions (U_i and E_j). Table 6b shows the factor loading of the questions in the adapted TAM. The threshold level for sufficient loading reported in the literature [18] is 0.7; the results for questions U_i to U_9 that represent usefulness load well with the first factor, thus we interpret this factor as “usefulness”, similar to one reported in [18].

| Table 6b: Factor Analysis for adapted TAM constructs. |
|------------------|------------------|------------------|
| Usefulness        | Ease of use      |
| Work more Quickly (U1) | 0.62 | -0.50 |
| Improve Performance (U2) | 0.68 | -0.51 |
| Increase Productivity (U3) | 0.71 | -0.40 |
| Effectiveness (U4) | 0.71 | -0.33 |
| Makes Job Easier (U5) | 0.73 | -0.17 |
| Useful (U6) | 0.73 | -0.26 |
| Easy to Learn (E1) | 0.41 | 0.55 |
| Easy to Perform (E2) | 0.40 | 0.59 |
| Clear and Understandable (E3) | 0.56 | 0.41 |
| Easy to become Skilful (E4) | 0.55 | 0.53 |
| Easy to Remember (E5) | 0.40 | 0.58 |
| Easy to Use (E6) | 0.53 | 0.55 |

The second factor would then be expected to represent “ease of use”; however, the factor loadings of E1 to E6 do not support this expectation as the factor is negatively correlated to “usefulness” questions and the factor is of similar strength to “usefulness” for the “ease of use” questions. According to the theory of reasoned action usefulness and ease of use are strongly correlated to self-predicted future (S1), the intention of actually using the technological innovation, if it is available. Table 6c and 6d show the factor correlation and significance of correlation among these 3 measures: Self-predicted future use is indeed significantly correlated to both usefulness and ease of use, while usefulness and ease of use have some, but not significant, correlation, which is consistent with reports in literature [25].

| Table 6c: Factor Correlation. |
|------------------|------------------|------------------|
| Usefulness        | Ease of Use      | WillUseIt        |
| Usefulness        | 1                | 0.24             | 0.57 |
| Ease of Use       | 0.24             | 1                | 0.37 |
| WillUseIt         | 0.57             | 0.37             | 1 |

Data analysis revealed p-values for usefulness and self-predicted future usage (p<0.001 for WillUseIt) at a significance level of 0.05. Concerning “ease of use” we identified significance regarding WillUseIt (p<0.001).

| Table 6d: Significance of Factor Correlation. |
|------------------|------------------|------------------|
| p-value | Usefulness | Ease of Use | WillUseIt |
| Usefulness | - | 0.010(s) | < 0.001(s) |
| Ease of Use | 0.010(s) | - | < 0.001(s) |
| WillUseIt | < 0.001(s) | < 0.001(s) | - |
These results show that usefulness and ease of use are important indicators for self-predicted future usage. We observed a relation between both variables and self-predicted future usage (WillUseIt).

7. Discussion

The empirical study investigated the technology acceptance of a generic groupware tool’s suitability for supporting the software evaluation process. This section discusses the results with related works and possible reasons for the outcomes.

Research issue 1 “Usefulness and ease of use”. As was expected from the tool selection process and the background of the study participants a majority of the participants found the collaborative tool, LiveNet, quite useful and easy to use for supporting collaborative tasks like architecture evaluation. This result is quite encouraging in terms of our intention to tailor/extend LiveNet to support the software architecture evaluation process.

Research issue 2 “Self-predicted future use”. A majority of the respondents was very positive about the regular use of the tool for collaborative tasks in the future, which supports the relationships put forward by the theory of reasoned action [25][26].

Research issue 3 “Preference of meeting style”. Most participants preferred a face-to-face meeting to a distributed arrangement using a groupware system. There can be several possible reasons for this outcome: LiveNet was not specifically developed for the software architecture evaluation process. Thus the tool needed the user to think about the process and how to perform it with the tool. Moreover, this finding is consistent with the results of our two controlled experiments, where participants also preferred face-to-face meetings despite the quality of the output of groupware supported meetings was better than the output of the face-to-face meetings [14][15].

Another reason might be that the participants of our study worked in groups without having a prior history or anticipation of future work as group. These factors do not usually allow any adaptive structuring of technology that might be a vital factor in improving participants’ satisfaction with a groupware-supported process [1]. The phases through which a group progresses have significant influence on group member behaviour, task focus, and overall performance [1] should be explored in further research on groupware-supported processes in software engineering in general and in software architecture evaluation process in particular. Moreover, the participants might not have been able to appreciate the time and resources that can be saved by reducing the travel requirement in the context of global software development as the study context did not have any traveling component.

The participants used two different meeting styles for generating scenarios (see Section 2.1). Such task usually involves consensus building, which is easy to achieve in a face-to-face arrangement. However, consensus building in a distributed arrangement requires a sophisticated tool like [9], while LiveNet does not provide such support. Moreover, the participants also identified several tasks that are not well supported by the current version of LiveNet. For example, real-time communication, decision-making for scenario generation and prioritization, and tradeoff analysis. We believe that a better support for these tasks by LiveNet would help raise the acceptance of distributed meetings for performing architecture evaluation tasks.

Research issue 4 “Tasks supported well or poorly by LiveNet”. Table 5 presents the findings about the tasks well or poorly supported by LiveNet; and it has already been mentioned that being a generic collaborative tool, LiveNet provides a good support for generic collaborative tasks, however, it does not support well tasks specific to the software architecture evaluation process.

Research issue 5 “Subjects experience and TAM responses”. Surprisingly, the study found only few notable differences in attitude towards LiveNet’s support for software architecture evaluation between subject groups with low, medium, or high experience in certain aspects of collaborative work. A reason may be the relatively homogeneous sample of subjects and the thorough preparation and teaching of the activities in the experimental context.

Apart for the issues identified for the reported research, we also analyzed the data for the “TAM reliability and factor validity evaluation”. The relationships between detailed sets of questions in TAM’s parts and higher-level measurement constructs “perceived usefulness”, and “perceived ease of use”, i.e., reliability measures supported the sets of questions used in TAM. This finding is in line with [25]. Factor analysis revealed a mixed picture; the factor analysis found two major factors, one of them was clearly associated with usefulness. However the second factor was definitely not ease of use. This may indicate a confusion of participants whether to rate tool support for general collaborative tasks or for distributed scenario meetings, for which they found the tool much less efficient and thus less useful. This finding needs more research, possibly with a replication of this empirical study.

8. Conclusion and Further Work

We have proposed a groupware-supported distributed architecture evaluation process aimed at supporting geographically dispersed stakeholders. Our proposed process is expected to address a number of logistical issues that characterize current architecture evaluation approaches by taking advantage of groupware technologies. We also need to identify and tailor a suitable groupware tool that is capable of supporting the different tasks of the proposed process. In this paper we have studied the user acceptance
of a generic groupware tool’s support for software architecture evaluation process. We used an adapted version of a well-known technology acceptance model, TAM, for gaining understanding of the participants’ attitude towards a groupware’s support for different tasks of architecture evaluation process. Main results from analyzing the data gathered using TAM were:

1. A majority of the participants found LiveNet quite useful and easy to use for supporting collaborative work involved in architecture evaluation.
2. A majority of the respondents was also very positive about the regular use of LiveNet for collaborative tasks in the future.
3. However, many participants prefer traditional face-to-face meetings to groupware supported distributed meetings.
4. Prior experience of the subjects had only little impact on their attitude towards the usefulness and ease of use of LiveNet.
5. TAM based results are reliable and consistent with the informal feedback from the study participants.

**Lessons learned.** Overall, the results show that LiveNet is a useful and easy to use tool for supporting collaborative tasks. However, it needs to be appropriately tailored/extended before providing an effective and efficient support for the software architecture evaluation process. The acceptance of groupware-supported processes by the users faces considerable challenges, possibly due to lower efficiency of the process as a result of missing suitable tool support.

**Future work** should pursue the following research directions:

1. Development of specific groupware functionality for distributed software architecture evaluation process. New processes and appropriate support mechanism will need investigation in empirical studies such as the one reported in this paper.
2. Investigate and model in more detail the relationships between process success of participants and their tool perception. This could allow to identify important factors that influence tool acceptance already during the collaborative process and allow adjustment of the process or support, if needed.
3. Another line of research is to investigate whether changes in the process can be better supported with existing tools, e.g., if the group outcome of the teams in the study is not significantly better than the result of so-called nominal team, i.e., teams that do not meet but where a moderator integrates the individual contributions.

**Acknowledgement** – We thank the participants of this experiment. The first author was working with the National ICT Australia when this was carried out.

**References**