The Impact of Systems Development Expertise on Information Systems Development Methodology Use

David Henderson
Virginia Tech

Follow this and additional works at: http://aisel.aisnet.org/amcis2006

Recommended Citation
http://aisel.aisnet.org/amcis2006/46

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2006 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
The Impact of Systems Development Expertise on Information Systems Development Methodology Use

David L. Henderson
Virginia Tech
davidlhenderson@vt.edu

ABSTRACT
Prior research on Task Technology Fit Theory (TTF) has found that task experience does not moderate perceptions of task-technology fit and does not directly affect technology usage. Additionally, researchers using TTF must develop new items to operationalize TTF constructs for each study. This study will reexamine the role of task experience in TTF within the context of systems development methodologies. Proposed results will indicate that task experience moderates perceptions of task-technology fit and influences intention to use. Furthermore, this study will assess perceptions of task-technology fit by employing a context-independent measure, thereby reducing the need to develop new task-technology fit items for each study.

Keywords
Systems development methodology, Task Technology Fit

INTRODUCTION
Task experience is an important variable for technology utilization (MacKay and Elam, 1992). However, research applying task experience to TTF suggests that task experience does not moderate perceptions of task-technology fit and that task experience does not directly affect computed-aided software engineering (CASE) tool usage (Dishaw and Strong, 2003). Researchers have not investigated the effect of task experience on technology utilization and on perceptions of task-technology fit within the context of systems development methodologies. Methodologies necessitate a larger change to daily activities than CASE tools (Hardgrave, Davis and Riemenschneider, 2003; Orlikowski, 1993; Roberts, Gibson, Fields and Rainer, 1998; Riemenschneider, Hardgrave and Davis, 2002). Thus, task experience may have different effects on perceptions of task-technology fit and on usage within the context of methodologies.

One problem with measures of task-technology fit is that they must be adapted for different tasks and technologies (Dishaw and Strong, 1998a). Another potential way to measure task-technology fit is to calculate the frequency of adaptations. Tracking the number of adaptations made to a technology represents a context-independent measure of fit that reduces the need to customize items for each study.

This study will investigate three research questions: 1) Does task experience moderate perceptions of task-technology fit within the context of methodologies? 2) What is the impact of task experience on methodology use? 3) Does frequency of adaptations represent another measure of task-technology fit?

LITERATURE REVIEW
TTF considers the intersection between task characteristics, individual characteristics, and technology characteristics (Goodhue, 1995; Goodhue and Thompson, 1995) and suggests a better fit among these constructs will result in higher use and better performance (Goodhue, 1995; Goodhue and Thompson, 1995). Figure 1 presents the TTF model (Goodhue and Thompson, 1995).

Given that Goodhue and Thompson (1995) maintained that TTF is intended to understand the general impacts of an entire set of systems, policies, or services and that TTF has been applied in a software development context (Dishaw and Strong, 1998a, 1998b, 1999, 2003), TTF constitutes an appropriate theoretical lens for understanding methodology use. When applying TTF to methodology use, the methodology represents the technology and the series of steps involved in systems development represents the task.
Differences in individual characteristics affect perceptions of task-technology fit and are often included as a moderating variable in TTF (Goodhue, 1995; Goodhue and Thompson, 1995; Dishaw and Strong, 1998a, 1999, 2003). Prior research has established a positive correlation between experience with a particular technology and higher use of that technology (Dishaw and Strong, 2003; Thompson, Higgins and Howell, 1994). Individual characteristics in TTF, operationalized as tool experience, moderate perceptions of task-technology fit (Goodhue, 1995; Dishaw and Strong, 2003).

Prior research has found that users must also possess task experience to effectively use a particular technology (MacKay and Elam, 1992). Dishaw and Strong (2003) found that task experience does not moderate perceptions of task-technology fit and that task experience does not have a main effect on CASE tool usage. Additional research in other contexts is necessary to clarify the role of task experience in perceptions of task-technology fit and on technology usage (Dishaw and Strong, 2003).

RESEARCH MODEL AND HYPOTHESES DEVELOPMENT

Figure 2 presents the research model. A systems development methodology is defined as “a coherent and systematic approach...which will guide developers on what steps to take, how these steps should be performed and why these steps are important in the development of an information system” (Fitzgerald, Russo and Stolterman, 2002, p. 5). Although prior research found that task experience did not affect perceptions of task-technology fit for CASE tools (Dishaw and Strong, 2003), many differences exist between CASE tools and methodologies. Compared to CASE tools, methodologies require a much larger change in developer behavior and daily activities (Hardgrave et al., 2003; Orlikowski, 1993; Roberts et. al., 1998; Riemenschneider et al., 2002). Furthermore, CASE tools can be used independently of a methodology (Vessey and Glass, 1998; Riemenschneider et al., 2002). Given these differences, I propose that task experience will affect perceptions of task-technology fit for methodologies.

Experience transforms factual knowledge into episodic knowledge (Kolodner, 1983). Episodic knowledge contains not only factual knowledge about concepts, tools, and techniques, but also contains knowledge, acquired over years of experience, for...
how to solve certain problems (Kolodner, 1983). Psychological research suggests experts possess greater episodic knowledge than novices (Boshuizen and Schmidt, 1992; Kolodner, 1983; Schenk, Vitalari and Davis, 1998). Greater episodic knowledge, acquired through years of experience, provides experienced developers with a better understanding of the tasks involved in developing a system.

On the other hand, psychological research suggests that novices possess factual knowledge, including the latest tools and techniques, but limited episodic knowledge (Kolodner, 1983). Novices lack trial and error experiences in real, domain-relevant situations, thereby limiting the external validity of their knowledge (Boshuizen and Schmidt, 1992; Kolodner, 1983; Schenk et al., 1998). Thus, inexperienced software developers possess factual knowledge about methodologies, but lack real-world experience developing systems.

H1: Developers with high levels of task experience will perceive higher task-technology fit than developers with low levels of task experience when the methodology matches the development context.

H2: Developers with low levels of task experience will perceive higher task-technology fit than developers with high levels of task experience when the methodology does not match the development context.

Prior research has not investigated whether task experience moderates the relationship between perceptions of fit and technology use. Task-technology fit may represent a salient explanatory variable for intention to use for some levels of task experience, but not others. Methodology use constitutes a context for which task experience may have a moderating effect on the link between perceptions of task-technology fit and intention to use.

Although one study (Leonard-Barton, 1987) indicated that experienced developers follow methodologies more rigorously than inexperienced developers, recent field research suggests that more experienced developers are less likely to rigorously follow a methodology (Fitzgerald, 1997; Kautz, Hansen and Jacobsen, 2004; Kozar, 1989; Lee and Kim, 1992; Nandhakumar and Avison, 1999). Although experienced developers do not rigorously follow formalized methodologies, they do acknowledge their importance (Beynon-Davies and Williams, 2003; Fitzgerald, 1997). Field research indicates that experienced developers are more likely to use a methodology tailored to the development context (Fitzgerald, 1997; Fitzgerald et al., 2002), indicating that experienced developers use methodologies strongly typed for a given development context.

Psychology research suggests that novices, because they lack episodic knowledge to solve a problem, rely on weak methods (Newell, 1969; Schenk et al., 1998; Voss and Post, 1988). Weak methods are general approaches that can be applied to many problems, but are inadequate for managing complex problems (Schenk et al., 1998; Vessey and Glass, 1998). Most methodologies recommend universal, general approaches to systems development, and can thus be construed as weak methods (Glass, 2004; Vessey and Glass, 1998). These findings suggest that inexperienced developers will use weak methods. Compared to experienced developers, inexperienced developers are more likely to follow a methodology rigorously and express a need for explicated methodologies to guide them through the development process (Fitzgerald, 1997; Kautz et al., 2004). Additionally, inexperienced developers are prone to goal displacement, a phenomenon in which the developer follows the methodology at the expense of actual development (Fitzgerald, 1998). Given these findings, I contend experienced developers will exhibit a higher intention to use a methodology than inexperienced developers when the methodology matches the development context.

H3: High task-technology fit will lead to a higher intention to use a methodology for developers with high levels of task experience than for developers with low levels of task experience.

Rather than following methods, experienced software developers omit steps of a methodology they deem unnecessary and change the methodology based on their perceptions of the development context (Beynon-Davies and Williams, 2003; Fitzgerald, 1997, 2000; Fitzgerald et al., 2002; Goulielmos, 2004). Compared to experienced developers, inexperienced developers are less likely to tailor a methodology to the development context (Fitzgerald, 1997; Kautz et al., 2004). Accordingly, I contend experienced developers will make more adaptations to a methodology than novices.

H4: The number of adaptations to a methodology will be higher for developers with high levels of task experience than for developers with low levels of task experience.

Researchers have operationalized task-technology fit using either Likert-scale questions or by calculating an interaction between task characteristics and technology characteristics (D’Ambra and Rice, 2001; D’Ambra and Wilson, 2004a, 2004b; Dishaw and Strong, 1998a, 1998b, 1999, 2003; Gebauer and Shaw, 2004; Goodhue, 1995, 1998; Goodhue et al., 2000; Goodhue et al., 1997; Goodhue and Thompson, 1995; Klaus et al. 2003; Klopping and McKinney, 2004). One problem with measuring TTF via Likert-scale questions or by calculating an interaction is the items must be customized for different tasks and technologies (Dishaw and Strong, 1998a).
Another potential way to measure task-technology fit is to calculate the frequency of adaptations. Provided the technology can be adapted, measuring perceptions of task-technology fit by number of adaptations represents a context-independent measure of task-technology fit. Furthermore, measuring task-technology fit via the number of adaptations reduces the need to customize questions for every task-technology fit study.

Rational developers will only change a methodology to make it more compatible with the development context. Similarly, if a methodology already matches the development context, rational developers will make fewer changes. Thus, the number of adaptations should be higher for a methodology that does not match the development context than for a methodology matching the development context.

H5: The number of adaptations to the methodology will be negatively correlated with perceptions of task-technology fit.

RESEARCH DESIGN

Independent and Dependent Variables

For hypotheses one and two, the dependent variable will be perceptions of task-technology fit and the independent variable will be the level of task experience. For hypothesis three, the independent variables will be perceived task-technology fit and level of task experience; the dependent variable will be intention to use. Performance was not chosen as a dependent variable because many other variables besides task-technology fit affect individual performance (Dishaw and Strong, 1999). I chose intention to use a methodology, rather than actual use, as the dependent variable, because TTF explains more variance in intention to use than actual utilization (Dishaw and Strong, 1999b) and because prior research shows a strong correlation between intention to use and actual use (Taylor and Todd, 1995; Venkatesh, Morris, Davis and Davis, 2003). For hypothesis four, the dependent variable will be the number of adaptations and the independent variable will be level of task experience. I will calculate a correlation between the number of adaptations made and perceptions of task-technology fit to test hypothesis five.

Sample

The sample will consist of approximately 30 experienced software developers and 30 inexperienced software developers. Task experience will be measured by job title and years of experience.

Design and Procedure

I will develop two scenarios; Table 1 outlines the experimental design. I will employ a 2x2 between-subjects experimental design in which experienced developers and inexperienced developers will be randomly assigned to each scenario; I will hold the development context constant across each scenario. Subjects in the high task experience condition (experts) will have ten or more years of relevant task experience (Ericsson and Lehmann, 1996); subjects in the low task experience condition (novices) will have one to two years of relevant task experience. The first scenario will consist of a description of a fictitious large-scale system followed by the description of a methodology designed to accommodate a large-scale systems development effort. The second scenario will describe the same fictitious large-scale system followed by the description of a methodology ill-suited for a large-scale development effort. After reading the scenario, subjects will be asked to respond to a series of Likert-scale and open-ended questions. I will pre-test and pilot test the scenarios with both experienced and inexperienced software developers.

<table>
<thead>
<tr>
<th></th>
<th>Fit</th>
<th>No Fit</th>
</tr>
</thead>
<tbody>
<tr>
<td>High Task Experience</td>
<td>Cell A: High TTF</td>
<td>Cell B: Low TTF</td>
</tr>
<tr>
<td>Low Task Experience</td>
<td>Cell C: High TTF</td>
<td>Cell D: Low TTF</td>
</tr>
</tbody>
</table>

Table 1. Outline of Experimental Design

Control Variables

Experience with a particular technology exhibits a positive correlation with utilization (Dishaw and Strong, 2003; Thompson et al., 1994) and moderates perceptions of task-technology fit (Goodhue, 1995; Dishaw and Strong, 2003). Accordingly, I will include tool experience as a covariate.

Data Analysis

To test hypotheses one and two, I will conduct a t-test of cells A and C and of cells B and D respectively. Significant differences for these two tests (p<.05), imply experienced developers better understand when a methodology fits the development context.
To test hypothesis three, I will conduct a two-way analysis of covariance (ANCOVA), with level of task experience and level of task-technology fit as the independent variables and intention to use as the dependent variable. A significant F statistic (p<.05) for the interaction between level of task experience and task-technology fit will indicate differences for intention to use. I will also conduct a series of planned comparisons (Buckless and Ravenscroft, 1990). In the high task-technology fit condition, I anticipate significantly higher means for intention to use for subjects with high levels of task experience than for subjects with low levels of task experience.

To test hypothesis four, I will conduct a t-test of cells B and D (p<.05). Significant differences would imply that experienced developers make more adaptations to a methodology than inexperienced developers. A significant negative correlation (p<.05) would signal that frequency of adaptation is significantly inversely related to perceptions of task-technology fit, supporting hypothesis five.

CONCLUSION

In conclusion, this study will offer several contributions. First, the anticipated results of this study will show that task experience interacts with perceptions of task-technology fit, implying that task experience influences perceptions of task-technology fit within the context of methodology use. This result would suggest that researchers include not only tool experience, but also task experience, as possible operationalizations of individual characteristics in TTF. The proposed findings will also illustrate an interaction between task experience perceptions of task-technology fit, suggesting that task-technology fit has a stronger effect on intention to use for experienced developers than for inexperienced developers. This result would imply that researchers consider the effects of task experience when testing TTF. Finally, this study will also provide another potential measure of task-technology fit—frequency of adaptations.

REFERENCES


