Evaluating the adoption potential of design science efforts: The case of APSARA

Sandeep Purao
College of Information Sciences and Technology
The Pennsylvania State University
University Park, PA 16802
Email: Sandeep-purao@psu.edu

Veda C. Storey
J. Mack Robinson College of Business
Georgia State University
P.O. Box 4015, Atlanta, GA 30302
Email: vstorey@gsu.edu

RESEARCH NOTE

Forthcoming in Decision Support Systems

Please contact the first author for citation information

Acknowledgements
The authors wish to thank the editor and reviewers for their comments on this paper. They also thank Arun Rai for his comments on earlier drafts, Detmar Straub for his counsel on research design, and Cecil Chua for his help in the data analysis. This research was supported by Georgia State University.
Evaluating the adoption potential of design science efforts:
The case of APSARA

Abstract

Improving information systems design outcomes requires, not only innovations in tools and methodologies, which are design science efforts, but also willingness on the part of potential developers to incorporate these innovations into their design practice. This is particularly true for reuse-based design, because of developers’ reluctance to stray from their established modes of work. This research argues that the technology acceptance model (TAM) may be used to assess whether potential systems developers will adopt IT innovations that facilitate reuse-based conceptual design. We use the core technology-acceptance model to assess intent to use, based on the constructs of ‘ease of use’ and ‘usefulness.’ One additional construct, compatibility, ensures that a key obstacle, minimizing changes to current design practice, is accounted for. We use a concrete instance of a previously proposed reuse-based design approach, APSARA, implemented in a prototype, as the basis for the study. The results indicate that significant contributors to the developers’ willingness to adopt reuse-based design approaches include ‘compatibility’ with current practice and ‘ease of use’ of the reuse-based design approach. We discuss implications of these findings for using tools such as APSARA, and for the use of TAM as an evaluation mechanism for design science efforts.

Keywords:
Reuse, Conceptual Design, APSARA, Technology Acceptance
1. Introduction

Reuse, which involves reusing products of previous software projects\(^1\) [Mili et al. 1995], represents the most potent approach to addressing the software development backlog [Jacobson et al. 1997, Bosch 2000, Szyperski 1999]. Reuse, however, is difficult to incorporate into practice because of the resistance from designers to adopt knowledge outcomes that were ‘not-invented-here’ [Joel 2002], particularly at the conceptual design stage of the system development process, which tends to be unstructured and creative. Although reusable artifacts such as analysis patterns [Coad et al. 1995] are readily available, developers rarely exploit these during the conceptual design of new applications. The reluctance to employ reuse-based design approaches in general, and for conceptual design in particular, can be traced to the perceived costs associated with reuse. If the perceived cost of reuse exceeds that of creating anew, the latter is preferred [Rothenberger and Dooley 1999, Rothenberger 2003]. These ‘costs’ of reuse are considered significant by the developers when formal reuse\(^2\) involves non-trivial adjustments to design practices [Frakes and Terry 1996, Mili et al. 1995, Morel and Alexander 2003]. Requiring such changes can prevent developers from accepting such reuse practices.

Reuse-based design approaches (e.g. [Batra and Davis 1992, Purao 1998, Purao et al. 2003]), which can be classified as design science efforts [Hevner et al. 2004] are, therefore, often aimed at lowering these human obstacles to reuse. In this paper, we argue that a possible mechanism to evaluate a key property of such approaches — adoption potential — is the technology acceptance model (TAM) [Davis 1989]. We employ three constructs from this research stream: perceived usefulness, ease of use, and compatibility with current practice. The first ensures that developers value the contribution of reuse to the design outcomes. The second ensures that developers do not need to expend significant effort to adopt the reuse-based design approach. The third ensures that the adoption of formal reuse does not require significant changes to current design practice. The rich stream of research on the technology acceptance model [Chin and Gopal 1995, Nance and Straub 1996] demonstrates

\(^1\) The definition by Mili et al. [1995] includes reuse of processes deployed to produce these artifacts. In this research, we focus on artifact reuse.

\(^2\) Human problem-solving naturally involves informal reuse [Lenat et al. 1990].
how the model can explain technology adoption by individuals and organizations. Hardgrave and Johnson [2003] suggest that the model may be used in a predictive manner to understand factors that contribute to adoption of systems development methodologies in general. In this research, we employ the model in a similar manner, to predict the intentions of developers for adopting specific reuse-based conceptual design tools by measuring the antecedents of usefulness, ease of use, and compatibility. A previously developed reuse-based conceptual design approach, APSARA [Purao and Storey 1997; Purao 1998], provides a design science artifact that forms the basis for developing the arguments. A prototype of APSARA is used as the basis for conducting the study described in this paper.

The objective of this research, thus, is to investigate the intentions of systems developers to adopt APSARA, a tool that facilitates reuse-based conceptual design. By doing this, we attempt to demonstrate how the technology-acceptance model may be used as an evaluation mechanism for outcomes from design science efforts.

The reminder of the paper is divided into five sections. Section 2 outlines background for the research, including an overview of design science efforts for reuse-based conceptual design, and a review of research related to the technology acceptance model (TAM). Section 3 develops the research model following TAM, describes the design science artifact used as the basis for the study, and outlines the study procedure. Section 4 describes the results highlighting key interpretations based on the constructs of usefulness, ease of use and compatibility along with intention to adopt. Section 5 discusses the implications for reuse-based design tools and for using TAM as an evaluation mechanism for design science research.

2. Background and Related Research

This section draws on prior work that informs the development of APSARA, a design science approach for facilitating reuse-based design [Purao and Storey 1997, Purao 1998], evaluation approaches for design science efforts, and the technology acceptance model (TAM) as a possible evaluation mechanism.

2.1 Obstacles to reuse-based design

Design with formal reuse involves making use of prior knowledge, often created by
others in situations analogous to, although not necessarily identical to, the problem at hand. Analogy making, therefore, is a fundamental cognitive process that aids in the reuse process. Analogy making assumes an intimate understanding of the source domain, the target domain, and a comparison of their structures and features in order to apply the knowledge from the former to the latter [Gentner 1983]. Analogy making can be difficult for developers [Irwin 2002] because it requires understanding a solution created by someone else, and adapting it to the problem at hand [Wohed 2000, Johannesson and Wohed 1999]. For non-trivial problems, analogy-making must be supplemented with the assembly of partial solutions using a compositional\(^1\) approach [De Sutter et al 2002] i.e. the adapted portions of prior knowledge must be integrated to create a new design [Zave and Jackson 1993, Gaedke and Rehse 2000, Garey and Johnson 1979, Mili et al 1995, Ramesh and Raghav Rao 1994]. The expected result of this process is a design produced at the conceptual design phase of an object-oriented application.

### 2.2 Approaches to reuse-based design

Design science [Hevner et al. 2004] efforts to overcome these obstacles are found in research on automated conceptual modeling [Bubenko and Wangler 1992], which formalizes approaches that assist developers in overcoming obstacles to reuse by, for example, generating the conceptual specification of an application from the users’ requirements, stated as natural language assertions [Lowry and McCartney 1991]. The assertions can be used to build a conceptual schema incrementally, through an interactive dialog with the developer. Tools have been developed that act as assistants to the developer, keeping track of details and making suggestions based upon embedded knowledge [Kwon and Park 1996, Lloyd-Williams 1997, Storey et al 1997, Morel and Alexander 2002]. The process that most automated approaches use is a linear sequence of tasks with user requirements as the starting point [Wohed 2000]. These approaches struggle to achieve an expert level, even for logical design, and rarely come close to representing expertise for conceptual design [Storey et al 1995]. Expert designers, however, employ behaviors not considered by these approaches. They categorize problem descriptions into standard abstractions [Batra and Davis 1992], apply contextual rules [Ericsson and Smith 1991, Cole 1988], use abstractions of real world

---

\(^1\) As opposed to generative, which ‘creates’ new artifacts using a set of synthetic rules (e.g. report generator).
situations [Cole 1988] and pattern-oriented mental models [Rouse and Morris 1986, Storey et al 1995], and reason by analogy [Prietula and Simon 1989]. An approach that incorporates support for such behaviors is proposed by Purao and Storey [1997], and Purao [1998].

2.3 The APSARA methodology\(^1\) for conceptual design with analysis patterns

This research uses as an exemplar the approach suggested by Purao and Storey [1997] and Purao [1998]. The approach simulates analogy-making and design by assembly during reuse-based design following a hybrid process that combines top-down and bottom-up development. Concepts from natural language requirements are used to decompose the requirements into smaller chunks (top-down) and reuse analysis patterns to constrain and guide this decomposition (bottom-up). The resulting process contains the phases: 1) retrieval of appropriate patterns, 2) instantiation of the patterns for the problem, and 3) synthesis of the instantiated patterns. Figure 1 shows the hybrid design process, as an extension of the process used in automated conceptual modeling [Wohed 2000]. The approach uses heuristics meant to support reuse-based design, grouped into three phases: Retrieval, Instantiation and Synthesis. The methodology has been implemented as a research prototype [APSARA 2005].

2.4 Evaluation of design science efforts

Evaluation is a crucial component of the design science process. Hevner et al [2004] describe two different, although overlapping approaches and corresponding reasons for

\(^1\) We use the term ‘methodology’ here to indicate a sequence of steps similar to an algorithm, instead of a
engaging in evaluation of design science efforts. The first posits evaluation as formative, i.e. as ‘essential feedback to the construction phase’ contributing to the generate-test cycle. The second posits evaluation as summative with approaches that include: observational, analytical, experimental, testing and descriptive [Hevner et al, 2004]. As described, neither approach addresses an essential evaluation element, the potential for adoption of the artifact that may result from the design science efforts. While this is implicit in two methodologies (field study and case study) suggested for the category ‘observational evaluation,’ there are several reasons why these two alternatives may be difficult to achieve in practice.

First, in several cases, the design science efforts result in artifacts that represent a ‘proof-of-concept’, which may not be sufficiently robust to insert into organizational environments [Nunamaker 2001]. Second, linkages with other components, necessary to insert the artifact in a technological complex, may not be in place; particularly if the effort represents a direction that is different and novel compared to current practice (consider, for example, the proposal for relational databases by Codd [1970]). Third, depending upon the focus on the design research effort, finding the target set of organizations that are receptive to change or potential users who are knowledgeable can require significant investments of time and efforts (See, for example, participative action research approaches such as Iversen et al. [2004] focusing on software process improvement).

A key concern, therefore, is how to evaluate design research outcomes, when the artifact cannot be immediately deployed in an organizational setting. While internal properties of the artifact may be assessed with analytical approaches, a useful alternative for external assessment is the technology acceptance model (TAM), which focuses on assessing the potential for adoption.

### 2.5 TAM as an evaluation mechanism for APSARA

The Technology Acceptance Model (TAM) [Davis 1989], hypothesizes that system use is affected by behavioral intentions, which are themselves affected by attitudes towards use. These, in turn, are affected by beliefs about the system, specifically, Perceived Usefulness (PU) and Perceived Ease of Use (EOU). It, therefore, suggests that the correlation of PU and EOU with system use explains why people may accept or reject a new technology. The

‘systems development methodology’ such as the object-oriented systems development methodology.
model draws on the theory of planned behavior/reasoned action [Fishbein and Ajzen 1975] to understand the rational choices individuals make in choosing to use or discard an innovation [Mathieson 1991]. The value of TAM as a model for explaining usage is evident in the consistency with which it has been applied [Davis 1989, Davis 1993, Davis et al 1989, Adams et al 1992, Chin and Gopal 1995, Nance and Straub 1996, Wu and Chen 2005]. Extensions to the core TAM model in recent years include TAM2 [Venkatesh and Davis 2000] as well as adaptations such as ISDAM [Hardgrave and Johnson 2003] that apply TAM2 to object-oriented systems development methodologies.

In spite of the consistent and frequent use of TAM to explain the acceptance or rejection of information technology innovations (Riemenschneider et al. [2002, pp. 1137-1139]), few have used TAM to predict the adoption potential of specific IT innovations. Such use of TAM is appropriate for our research because we are interested in evaluating the adoption potential of innovative reuse-based design tools such as APSARA\(^1\). Lacking an a priori reason for why TAM would not apply to assessing adoption intention\(^1\), we posit that TAM will allow us to predict the intention of developers to adopt APSARA. Because deploying APSARA in an organizational setting can require significant organizational commitment, our use of TAM is necessarily restricted. For example, we do not include constructs such as ‘subjective norm’ posited in the extended version (TAM2), and must rely on ‘potential’ instead of ‘actual’ users of APSARA. The next section elaborates on these choices and develops the research model.

3. **Research model and procedure**

To investigate potential developers’ intentions to adopt APSARA [Purao 1998], we use the core TAM model and devise an experimental procedure following prior research (e.g., Nance and Straub [1996], Chin and Gopal [1995]).

3.1 **The research model**

We consider an implementation of the reuse-based approach such as APSARA an IT

---

\(^1\) Riemenschneider et al. [2002] describe different theoretical models e.g. TAM, TAM2, PCI, TPB (a precursor to TAM, and MPCU. Based on the characteristics of the design science effort, it may be possible to use these for assessing the adoption potential.
innovation that facilitates reuse-based design (similar to a CASE tool that facilitates modeling). Here, it is necessary to make a distinction between a methodology and a tool. Fichman and Kemerer [1997] describe a process innovation as one that changes the way a job is performed. Hardgrave and Johnson [2003], drawing on this description and Ivari et al [1998], argue that adopting a process innovation requires a much more radical change than adopting tools or technologies [Orlikowski 1993]. APSARA represents a tool that requires minimal change to the way in which software design is done.

We measure the constructs in TAM that would allow us to predict whether the proposed IT innovation (APSARA) may be accepted or rejected by the user population, which consists of potential designers. We identify three constructs as determinants of intention to use. These include the two original constructs in TAM, perceived usefulness (PU), and perceived ease of use (EOU), and the compatibility (CO) construct added in later research. The choice is dictated by two factors that are crucial for adopting formal reuse: (a) reducing the effort a developer must undertake for reusing artifacts created by someone else; and (b) ensuring that reuse does not require new ways of working [Frakes and Terry 1996, Mili et al 1995]. Additional constructs such as trialability, visibility or image [Moore and Benbasat 1991], are, therefore, not considered relevant for the purpose of this research. The three constructs are defined below.

- **Perceived Ease of Use (PEU)** is “the degree to which a person believes that using a particular system would be free of effort” [Davis 1989, p. 320],
- **Perceived Usefulness (PU)** is “the degree to which a person believes that using a particular system would enhance his or her performance” [Davis 1989, p. 320].
- **Compatibility (CO)**, is “the degree to which an innovation is perceived as being consistent with the existing values, needs and past experiences of potential adopters” [Moore and Benbasat 1991]

Figure 2 shows the research model we used to assess developers’ intention to adopt (use) a reuse-based design approach following the technology-acceptance model. The

---

1 Our argument is similar to that by Rose and Straub [1998], who suggest extending TAM to general use of IT.
following hypotheses were expected to be true.

<table>
<thead>
<tr>
<th>Hypothesis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perceived usefulness will have a positive influence on Intent to adopt.</td>
</tr>
<tr>
<td>Ease of use will have a positive influence on Intent to adopt.</td>
</tr>
<tr>
<td>Compatibility will have a positive influence on Intent to adopt.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 1. Hypotheses Posited</th>
</tr>
</thead>
</table>

The constructs are operationalized following specific items summarized in Chin and Gopal [1995, p. 48]. The items used in the survey appear in Appendix III. Table 2 summarizes the constructs and corresponding items.

<table>
<thead>
<tr>
<th>Construct</th>
<th>Measure</th>
<th>Items</th>
<th>Type of Variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perceived Usefulness</td>
<td>Adapted from Davis 1989</td>
<td>2</td>
<td>Likert (7)</td>
</tr>
<tr>
<td>Ease of Use</td>
<td>Adapted from Davis 1989</td>
<td>4</td>
<td>Likert (7)</td>
</tr>
<tr>
<td>Compatibility</td>
<td>Adapted from Moore and Benbasat 1991</td>
<td>4</td>
<td>Likert (7)</td>
</tr>
<tr>
<td>Intent to Use</td>
<td>Adapted from Davis 1989</td>
<td>4</td>
<td>Likert (7)</td>
</tr>
</tbody>
</table>

| Table 2. Constructs and Measures                     |

3.2 The design research artifact used in the study

The implementation available in a prototype called APSARA (Automated Pattern Synthesis and Retrieval Assistant) [Purao 1998, Purao et al. 2003] was used for this research. The artifact was seen as a tool, i.e. a concrete, reified instance of the underlying methodology. The intent of the study, thus, was not to assess whether the developers would adapt their work to the APSARA methodology, but rather, whether they would incorporate the APSARA tool in their design practice. The example below shows how the APSARA methodology, as implemented in the APSARA tool, could be applied to a large set of requirements represented as a set of use cases.
The example contained three use cases\(^1\). The APSARA tool was applied to each use case separately. The resulting designs could then be combined manually to construct the overall design. Figure 3 illustrates use of the prototype. It shows a separate ‘package’ for each of the three use cases (shown in Appendix I) within Rational Rose [Rational 2004]. The largest class diagram, generated for use case 2, contained 11 classes and 10 relationships. Some objects were not instantiated (e.g., the ‘Actor’ object); others were instantiated differently (e.g., the ‘Participant’ object was instantiated as ‘clerk’ for use cases 1 and 3, and as ‘worker’ for use case 2). Figure 4 shows the progression from use cases to conceptual design with and without the use of the APSRA tool. The figure shows that the use of APSARA requires minimal changes to the task of constructing the conceptual design.

### 3.3. Study Procedure

The procedure used in this research was an enhanced version of the procedure suggested by Chin and Gopal [1995]. The subjects were shown a short written description of APSARA, followed by a demonstration of the research prototype (see Appendix II). The demonstration resulted in conceptual models of individual use cases. The subjects were then shown the merged model (mimicking the outcomes shown in the previous sub-section), and asked to respond to a questionnaire. The procedure was, thus, similar to a free simulation exercise, where subjects are asked to make decisions by simulating a real-world situation as part of the experiment [Fromkin and Streufert 1976]. Since there are no preprogrammed treatments, the experiment allows the values of independent variables to range over the natural range of the subject’s experience. In effect, the experimental tasks induce subject responses, which are then measured via the research instrument. [Gefen et al 2000, p. 12].

---

\(^1\) Another example containing two use cases was also used during the study.
Two examples were used for demonstration (see Appendix I for one of the examples). Data was gathered on multiple occasions from students, who were majoring in computer information systems or related disciplines, and completing specialized courses in object-oriented systems analysis and design at different public universities. A total of 69 respondents participated in the experiment over three separate occasions over a period of seven months. No compensation was provided to the subjects for participating in the experiment. Of the 69 respondents, 12 were dropped because they either did not complete the survey or gave inappropriate responses such as providing the same answer on all questions. The urban campus of the university, where the survey was conducted, meant that the students were non-traditional, exhibited a strong commitment to the IT profession with an average of 1.56 years of experience in industry, and were older than the younger, traditional students. Table 4 shows these demographic characteristics.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Descriptive Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age</td>
<td>Average: 23.3 Low: 19, High: 31</td>
</tr>
<tr>
<td>Gender</td>
<td>Male: 39, Female: 18</td>
</tr>
<tr>
<td>Experience</td>
<td>Average (number of years): 1.56</td>
</tr>
</tbody>
</table>

Table 4. Demographic characteristics of respondents (Based on 57 usable surveys)

There is simply no reliable way, given the current maturity of the IS field, to decide how representative our participants are of the larger population of developers (see similar arguments from Guindon [1990]). There is no standard type of individual who becomes a software developer. They can differ in education, experience and attitude. The participants had, on average some prior experience in the IS industry. Many were continuing work while obtaining a specialized degree in MIS. They were, thus, representative of at least some segment of the population of actual developers. It is, indeed, possible to argue that the results may be more robust if the sample were to include developers, who are surveyed or interviewed at their workplace. Our choice of the sample was, thus, dictated by access to a pool of potential developers, who were enrolled in a specialized degree in MIS.

4. Results

Initial descriptive statistics were computed to provide an indication of whether the
data confirms to the general notions captured by the items for each construct. Table 5 shows these statistics. The responses show that on the scale that varied from extremely likely (1) to extremely unlikely (7), all items exhibit a tendency towards agreement; that is, potential adopters indicate they are likely to use it (2.56 on a scale of 1-7), and they perceive it to be useful (2.74), easy to use (2.69), and compatible with their work (3.01).

<table>
<thead>
<tr>
<th>Construct</th>
<th>Item</th>
<th>Mean</th>
<th>S.D.</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intent to Use</td>
<td>IU1</td>
<td>2.3</td>
<td>0.829</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IU2</td>
<td>2.82</td>
<td>0.956</td>
<td></td>
</tr>
<tr>
<td>Perceived Usefulness</td>
<td>PU1</td>
<td>2.29</td>
<td>0.825</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PU2*</td>
<td>4.98</td>
<td>1.382</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PU3</td>
<td>2.46</td>
<td>0.934</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PU4*</td>
<td>4.82</td>
<td>1.309</td>
<td></td>
</tr>
<tr>
<td>Ease of Use</td>
<td>EU1</td>
<td>2.61</td>
<td>0.908</td>
<td>2.56</td>
</tr>
<tr>
<td></td>
<td>EU2</td>
<td>3.04</td>
<td>1.206</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EU3</td>
<td>2.55</td>
<td>0.913</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EU4</td>
<td>2.57</td>
<td>1.024</td>
<td></td>
</tr>
<tr>
<td>Compatibility</td>
<td>CO1</td>
<td>2.84</td>
<td>1.108</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CO2*</td>
<td>4.91</td>
<td>1.133</td>
<td>2.74</td>
</tr>
<tr>
<td></td>
<td>CO3*</td>
<td>4.89</td>
<td>1.330</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CO4</td>
<td>3.02</td>
<td>1.183</td>
<td>2.69</td>
</tr>
</tbody>
</table>

Legend: Asterisk indicates reverse worded items. Average adjusts for the reversed scale.

**Table 5. Descriptive Statistics (see Appendix I)**

Additional analysis was performed to understand whether the items map to the underlying constructs, and whether the causal relationships postulated in TAM can be predicted from the data gathered. The responses were factor analyzed fixing the number of factors at four. The principal component analysis was conducted with the Varimax method with Kaiser Normalization. Table 6 shows the rotated component matrix.
All items loaded as expected (shaded and bold-faced) except PU1 and PU3 (shaded but not bold-faced). These were, therefore, dropped from subsequent analysis. With the remaining items, the research model was tested using a multiple regression analysis. A test of this model yielded an adjusted R-squared of 33.9%; thus, it explained 33.9% of the variance in the dependent variable, Intent to Use (IU). An ANOVA was performed to test for significance. The F-test indicated that the result was significant. The results were significant at the alpha=0.05 level (p=0.000, F=10.583, df=56). Individual t-tests conducted on the constructs revealed that Ease of Use (EU) and Compatibility (CO) were significant predictors of Intent to Use (IU). The corresponding values were t=2.156, p=0.036 for Ease of Use (EU), and t=2.839, p=0.006 for Compatibility (CO). However, Perceived Usefulness (PU) was not found to be a significant predictor of Intent to Use (IU) with the values t=0.533, p=0.597. Table 7 summarizes these results.

<table>
<thead>
<tr>
<th></th>
<th>Unstandardized Coefficients B</th>
<th>Std. Error</th>
<th>Standardized Coefficients Beta</th>
<th>t</th>
<th>Sig.</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Constant)</td>
<td>2.386</td>
<td>.556</td>
<td>4.295</td>
<td>.000</td>
<td></td>
</tr>
<tr>
<td>PU</td>
<td>-4.829E-02</td>
<td>.091</td>
<td>-0.067</td>
<td>-0.533</td>
<td>.597</td>
</tr>
<tr>
<td>EU</td>
<td>.265</td>
<td>.123</td>
<td>.268</td>
<td>2.156</td>
<td>.036</td>
</tr>
<tr>
<td>CO</td>
<td>.306</td>
<td>.108</td>
<td>.399</td>
<td>2.839</td>
<td>.006</td>
</tr>
</tbody>
</table>

Table 7. Summary of Regression Results

The hypotheses posited as true following TAM were, therefore, partially supported. The antecedents posited in the causal model yielded the values expected (table 5); the causal relationships (table 7) among the constructs were supported for H2 and H3, but not for H1.

5. Discussion

The results provide interesting insights into the rationale software developers might employ in their decision to adopt tools such as APSARA that support formal reuse. The results of the survey show that potential developers exhibit a strong intention towards adoption. Two constructs appear to determine their intent to adopt. These are ‘ease of use’
and ‘compatibility.’ The ease of use construct refers to the extent to which a potential system developer expects the use of a reuse-based design approach to be free of effort. The compatibility construct refers to the degree to which the reuse-based design approach is perceived as being consistent with the existing design practice. These are greater contributors to the intent to use than perceived usefulness. Conceptual design is a difficult task. Although reusable artifacts such as analysis patterns are available, they have not lead to extensive reuse during conceptual design. A key obstacle to realizing this benefit has been the high perceived “cost of reuse” [Rothenberger et al 2003]. Prima facie, our results confirm the importance of overcoming this cost. It is, however, possible that the usefulness construct becomes significant if professional developers are used as subjects. Interestingly, the results are compatible with findings from Riemenschneider et al. [2002], who found that usefulness was not a significant construct in determining acceptance of object-oriented systems development methodologies.

The results are also true given what we know about resistance to reuse. Even when it has been demonstrated that reuse can potentially lead to significant benefits, developers are reluctant to engage in formal reuse, a syndrome sometimes referred to as “not-invented-here” [Joel 2002, Griss 1993, p. 553]. Studies repeatedly confirm that mere demonstration of benefits of reuse is not sufficient to entice developers to engage in formal reuse [Griss 1993, Mili et al 1995, Rothenberger et al. 1999, Rothenberger 2003]. The theme is echoed in studies related to knowledge work, where a significant obstacle to use of knowledge repositories has been adjustments to work practices that deters potential users from engaging with the knowledge repository, when faced with significant time pressure and availability of alternatives such as knowledge networks [Bansler and Havn 2001]. A possibility is to consider ‘attitude’ to investigate individual and organizational adoption of technology [Yang and Yoo 2004]. A second possible interpretation of the results may be that perceived usefulness is considered as a prerequisite for, but not a significant determinant of, the intent to adopt. This can be true when alternatives are available to a developer such as relying on his/her own expertise or informal networks [Bansler and Havn 2001] versus using a tool such as APSARA that facilitates reuse-based design. If all alternatives are perceived to be equally useful, the alternative involving few adjustments to work practices is likely to win.
The importance of minimizing disruptions to current work practices has also been emphasized in the specific context of reuse-based design by Frakes and Terry [1996], Mili et al. [1995], Morisio et al. 2002, and Ravichandran and Rothenberger [2003].

The results also suggest a novel role for tools such as APSARA. These tools may be useful to introduce developers to reuse-based design in a familiar (see ‘compatibility’) and non-demanding (see ‘ease of use’) manner. The lack of significance attached to perceived usefulness for deciding about adoption may mean that developers may treat these tools, not as a significant contributor to effective design outcomes, but rather, as a first step towards learning about reuse-based design. The significant influence of ‘compatibility’ and ‘ease of use’ on ‘intention to adopt,’ on the other hand, suggests that developers may value these tools as vehicles to learn about formal reuse. This is particularly true for conceptual design, which involves codification and abstraction of knowledge such as the analysis patterns that APSARA contains. The reuse of these forms of knowledge can require much learning that tools similar to APSARA can facilitate. Positioning the use of APSARA as learning can overcome the conundrum developers may face while attempting to reuse analysis patterns during conceptual design. As novices, the analysis patterns provide useful codifications of prior knowledge. However, they are difficult to understand and reuse. As experts, the developers may have already internalized the knowledge codified in the analysis patterns making tools such as APSARA superfluous. A novel role that tools such as APSARA can, therefore, play may be helping developers learn the craft of reuse-based design.

For research that focuses on TAM, the results provide different contributions. Much prior work on TAM, which has dealt with general IT applications (e.g. email [Karahana 1993], spreadsheet [Mathieson 1991], technology for the disabled [Goette 1995]), general IT usage [Rose and Straub 1998], or systems development methodologies [Hardgrave and Johnson 2003]. In contrast, our work focuses on knowledgeable users of IT (potential information systems developers) for a specific IT innovation (tool) that facilitates the IS development process but does not radically alter it. The results suggest that for the scenarios considered in this work, the constructs of Ease of Use and Compatibility are better predictors of Intent to Use than the much-emphasized constructs of Perceived Usefulness.

1 We appreciate the suggestions from an anonymous reviewer that lead to this interpretation.
or Relative Advantage [Hendrickson et al 1993]. Our results are consistent, though, with those for software development methodologies [Hardgrave and Johnson 2003], and provide greater specificity to the argument in Davis and Venkatesh [2004] in the context of evaluating design research efforts. Further studies are clearly necessary to shed light on the relative contribution of these constructs. A second contribution to the discourse on research on TAM is the manner in which we have used TAM in our study, namely to explore the adoption potential of an IT innovation. Few studies (e.g. [Rose and Straub 1998, Hardgrave and Johnson 2003]) have attempted to apply TAM in this manner. However, in Rose and Straub [1998], the constructs measured included actual use of IT, and in Hardgrave and Johnson [2003], two out of three respondents had used the object-oriented software development methodology. In contrast, in the study reported in this paper, none of the users had any experience with APSARA because it represents a design science effort that is not widely available. The study, thus, focuses on predicting the potential for adoption of an IT innovation.

The study we have outlined, and the results, suggest a possible role that models such as TAM can play for design science research. As Hevner et al. [2004] suggest, a number of criteria may be used to evaluate these efforts. One of these evaluation categories is “experimental” evaluation, which includes evaluating properties such as usability. They are, however, silent on other specific mechanisms that may be of value for evaluating the design science outcomes. A core argument in our paper has been the importance of assessing the willingness of potential adopters of the technology to actually use the technology. We have, therefore, suggested TAM as a possible mechanism to evaluate the outcomes of design science. A large number of design science efforts never proceed beyond the laboratory, ending the research cycle as a research prototype. In spite of calls to move the outcomes into the marketplace [Nunamaker 2001], academics find it difficult to invest the time required to do so. Convincing the research community and practitioners of the potential for adoption – with TAM or other models – can go a long way towards focusing greater attention to such outcomes.

Like all research, ours has its limitations. First, the participants in the study were recruited from a student population. Similar use of student subjects is, however, reported in
other studies (e.g. [Purao et al 2003, Gefen et al 2000]). Second, the result suggesting ‘compatibility’ as a contributor to intention to adopt may be questioned based on the use of student subjects. We recognize this as a limitation but stress that the population from which the sample was drawn consisted of non-traditional students, who had at least some industry experience. Further, the notion of compatibility maps to changes required to a design task, specifically that of constructing the conceptual design, which, arguably, does not change between students and practitioners, except at a cognitive level.

In summary, this research makes two key contributions. First, it identifies the constructs of ease of use and compatibility as significant contributors to the intention to adopt reuse-based design tools. In doing so, it shows that the adoption decision will require more than a demonstration of usefulness or relative advantage. Second, it shows application and use of the technology-acceptance model to assess the adoption potential of IT innovations. Clearly, larger scale studies, including introductions of innovations such as APSARA in the field, are needed to further explore these constructs. The study reported in this paper describes a first step in this direction.
References

Information Technology: A Replication. MIS Quarterly, 16(2), 227-247.
2. APSARA. 2005. Heuristics for automated pattern retrieval and synthesis. Available at
Addison Wesley.
Database for Advances in Information Systems. 26(2,3). May/August. Pp. 42-64.
Lake City. April.
Information Technology, MIS Quarterly, 13 (3), September, 319-339.
A Comparison of Two Theoretical Models, Management Science, 35 (8), August, 982-1003.
16. Davis, F. D. and Venkatesh, V., Toward Pre-prototype User Acceptance Testing of New
Information Systems: Implications for Software Project Management, IEEE Transactions on
systems, languages, and applications, Volume 37 Issue 11.
22. Fromkin, H. L., and Streufert, S. 1976. Laboratory Experimentation, in Handbook of Industrial and


Appendix I – Applying APSARA to a set of Use Cases

**Domain: Human Resources**

<table>
<thead>
<tr>
<th>Use Case 1: create work teams</th>
<th>Use Case 2: assign teams to projects</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 clerk enters new team</td>
<td>1 clerk enters a project</td>
</tr>
<tr>
<td>2 system displays available employees</td>
<td>2 system displays list of teams</td>
</tr>
<tr>
<td>3 clerk assigns each employee to a team</td>
<td>3 clerk selects a team</td>
</tr>
<tr>
<td>4 system records employees and teams</td>
<td>4 systems assigns the team to project</td>
</tr>
<tr>
<td>Use Case 3: record job progress</td>
<td>5 system updates team status</td>
</tr>
<tr>
<td>1 clerk selects a job</td>
<td></td>
</tr>
<tr>
<td>2 clerk records worker on a job</td>
<td></td>
</tr>
<tr>
<td>3 system updates job progress</td>
<td></td>
</tr>
<tr>
<td>4 system updates job status</td>
<td></td>
</tr>
</tbody>
</table>

**Class Diagram for Use Case 1**

**Class Diagram for Use Case 2**

**Class Diagram for Use Case 3**
Appendix II – Description and Demonstration

Description of Research on Reuse-based Design and the APSARA Prototype

Reuse-based System Development
Development of computer-based information systems requires significant efforts from human analysts and designers. Information technology support in the form of CASE Tools (e.g., Rational Rose) has been available for documenting work products of systems developers for some time now. Automating the process of software development, particularly in the early stages of conceptual design is, however, still difficult.

Research on Improving Reuse
One research project aimed at developing such support is being carried out at ABC State University. Currently, it exists, in a limited capacity, as a research prototype under the name APSARA: Automated Pattern Synthesis and Retrieval Assistant. It automates the process of reuse of Analysis Patterns suggested by Coad. Under the covers, it uses several techniques including text parsing, automated reasoning and intelligent heuristics.

Automated Pattern Synthesis and Retrieval Assistant
The tool, APSARA, automates the process of reuse-based design by providing computer-based support for three tasks: retrieval, instantiation and synthesis.

1. Retrieval – identification of appropriate analysis patterns based on natural language descriptions provided by the analysts, for example, as use cases.
2. Instantiation – making the retrieved analysis patterns specific to the problem being considered, based on the description contained in the use case.
3. Synthesis – connecting the instantiated patterns to create a preliminary class diagram for each use case.

These tasks are carried out as a sequence of steps, using clearly marked buttons and following instructions shown in the tool. To make the process transparent to the designer, there are several (more than three) steps in the tool. Currently, the tool does not allow the designer to intervene in the process. The outputs are generated in a text format that may then be imported into a CASE tool such as Rational Rose.

Using APSARA
Using APSARA, system designers may type a use case description into the tool, and follow the instructions to generate a preliminary class diagram for each use case. Since this is an automated tool that cannot possess ‘common-sense,’ errors are possible. The designer can check the class diagrams and make any adjustments as necessary. The designer’s job is, then, to combine the individual class diagrams to create the complete class diagram.

Demonstration
The demonstration will involve using APSARA to create class diagrams based on several use cases. Following the demonstration, the outputs will be shown as published from the CASE tool, where they are imported.

Feedback
Your feedback is sought following the demonstration. Your impressions and intentions about the tool will help move the research forward. Thank you for your participation!
### Appendix III – Survey Instrument

Assuming that any decision to use APSARA would be totally up to you, and APSARA was available:

<table>
<thead>
<tr>
<th>Construct and Item</th>
<th>Likely</th>
<th>Unlikely</th>
</tr>
</thead>
<tbody>
<tr>
<td>IU1</td>
<td>Extremely</td>
<td>Extremely</td>
</tr>
<tr>
<td>How would you rate your potential to use it?</td>
<td>Extremely</td>
<td>Extremely</td>
</tr>
<tr>
<td>IU2</td>
<td>Quite</td>
<td>Quite</td>
</tr>
<tr>
<td>Would you use it often?</td>
<td>Quite</td>
<td>Quite</td>
</tr>
<tr>
<td>PU1</td>
<td>Extremely</td>
<td>Extremely</td>
</tr>
<tr>
<td>I would find APSARA useful on the job.</td>
<td>Extremely</td>
<td>Extremely</td>
</tr>
<tr>
<td>PU2</td>
<td>Quite</td>
<td>Quite</td>
</tr>
<tr>
<td>Using APSARA would not enhance my effectiveness on the job.</td>
<td>Quite</td>
<td>Quite</td>
</tr>
<tr>
<td>PU3</td>
<td>Neither</td>
<td>Neither</td>
</tr>
<tr>
<td>Using APSARA on the job would increase my productivity.</td>
<td>Neither</td>
<td>Neither</td>
</tr>
<tr>
<td>PU4</td>
<td>Extremely</td>
<td>Extremely</td>
</tr>
<tr>
<td>Using APSARA would not improve my performance on the job.</td>
<td>Extremely</td>
<td>Extremely</td>
</tr>
<tr>
<td>EU1</td>
<td>Extremely</td>
<td>Extremely</td>
</tr>
<tr>
<td>I would find APSARA easy to use.</td>
<td>Extremely</td>
<td>Extremely</td>
</tr>
<tr>
<td>EU2</td>
<td>Quite</td>
<td>Quite</td>
</tr>
<tr>
<td>I would find it easy to get APSARA to do what I want to do.</td>
<td>Quite</td>
<td>Quite</td>
</tr>
<tr>
<td>EU3</td>
<td>Neither</td>
<td>Neither</td>
</tr>
<tr>
<td>Learning to use APSARA would be easy for me.</td>
<td>Neither</td>
<td>Neither</td>
</tr>
<tr>
<td>EU4</td>
<td>Extremely</td>
<td>Extremely</td>
</tr>
<tr>
<td>It would be easy for me to be skillful at using APSARA.</td>
<td>Extremely</td>
<td>Extremely</td>
</tr>
<tr>
<td>CO1</td>
<td>Extremely</td>
<td>Extremely</td>
</tr>
<tr>
<td>Using APSARA would be compatible with my own system design work.</td>
<td>Extremely</td>
<td>Extremely</td>
</tr>
<tr>
<td>CO2</td>
<td>Quite</td>
<td>Quite</td>
</tr>
<tr>
<td>Using APSARA would not be compatible with my current work on system design.</td>
<td>Quite</td>
<td>Quite</td>
</tr>
<tr>
<td>CO3</td>
<td>Neither</td>
<td>Neither</td>
</tr>
<tr>
<td>Using APSARA would not fit well with the way I like to carry out my system design work.</td>
<td>Neither</td>
<td>Neither</td>
</tr>
<tr>
<td>CO4</td>
<td>Extremely</td>
<td>Extremely</td>
</tr>
<tr>
<td>Using APSARA would fit my work style for system design.</td>
<td>Extremely</td>
<td>Extremely</td>
</tr>
</tbody>
</table>

**Notes:**
- Items PU2, PU4, CO2, and CO3 are reverse worded.
- Questions requesting demographic information are not shown.