國立中山大學九十一學年度博士班招生考試試題

科目：資訊科技論文評述第一節【資管系】

Read the attached paper and answer the following questions. Note that the time is limited, and you should budget your time carefully. You are suggested to spend 50 minutes in reading the paper and another 50 minutes in answering the questions.

1. Please describe the problem that the paper wants to solve. What are the features of a solution that the authors think should have?
2. Please describe the proposed approach in detail.
3. The proposed approach uses a standardized language to communicate between producers and consumers. What are the functions of the language. What is the key idea of designing such a language?
4. What additional features the system should have for user point of view? Please propose your solution.

A Cooperative Approach to Support Software Deployment Using the Software Dock

Richard S. Hall, Dennis Heimbigner, Alexander L. Wolf
Software Engineering Research Laboratory
University of Colorado
Boulder, Colorado 80309-0430 USA
[rickhall,dennis,alw]@cs.colorado.edu

ABSTRACT
Software deployment is an evolving collection of interrelated processes such as release, install, adapt, reconfigure, update, activate, deactivate, remove, and retire. The connectivity of large networks, such as the Internet, is affecting how software deployment is performed. It is necessary to introduce new software deployment technologies that leverage this connectivity. The Software Dock framework creates a distributed, agent-based deployment framework to support the ongoing cooperation and negotiation among software producers themselves and among software producers and software consumers. This deployment framework is enabled by the use of a standardized deployment schema for describing software systems, called the Deployable Software Description (DSD) format. The Software Dock also employs agents to traverse between software producers and consumers in order to perform software deployment activities by interpreting the descriptions of software systems. The Software Dock infrastructure allows software producers to offer their customers high-level deployment services that were previously not possible.

Keywords
Software deployment, Java, mobile agents, configuration management

1 INTRODUCTION

The connectivity of large networks, such as the Internet, is affecting how software deployment is being performed. The simple notion of providing a complete installation procedure for a software system on a CD-ROM is giving way to a more sophisticated notion of ongoing cooperation and negotiation among software producers and consumers. This connectivity and cooperation allows software producers to offer their customers high-level deployment services that were previously not possible. In the past, only software system installation was widely supported, but already support for the update process is becoming common. Support for other software deployment processes, though, such as release, adapt, activate, deactivate, remove, and retire [see Section 2] is still virtually non-existent.

ICSE '99 Los Angeles CA

New software deployment technologies are necessary if software producers are expected to accept more responsibility for the long-term operation of their software systems. In order to support software deployment, new deployment technologies must:

- operate on a variety of platforms and network environments, ranging from single sites to the entire Internet,
- provide a semantic model for describing a wide range of software systems in order to facilitate some level of software deployment process automation,
- provide a semantic model of target sites for deployment in order to describe the context in which deployment processes occur, and
- provide decentralized control for both software producers and consumers.

The Software Dock research project addresses many of these concerns. The Software Dock is a system of loosely coupled, cooperating, distributed components. The Software Dock supports software producers by providing the release dock that acts as a repository of software system releases. At the heart of the release dock is a standard semantic schema for describing the deployment requirements of software systems. The field dock component of the Software Dock supports the consumer by providing an interface to the consumer's resources, configuration, and deployed software systems. The Software Dock employs agents that travel from release docks to field docks in order to perform specific software deployment tasks while docked at a field dock. The agents perform their tasks by interpreting the semantic descriptions of both the software systems and the target consumer site. A wide-area event system connects release docks to field docks and enables asynchronous, bi-directional connectivity.

The purpose of this paper is to discuss how the Software Dock project supports software deployment processes. This is accomplished by first introducing the processes that

comprise software deployment in Section 2. Section 3 provides a high-level introduction to the Software Dock, a framework for software deployment, while Section 4 describes the Deployable Software Description (DSD) format, a critical piece of the Software Dock project used to describe the deployment requirements of software systems. Section 5 discusses specific deployment process support through the use of agents. Section 6 discusses security as it relates to the deployment and the Software Dock specifically, while Section 7 discusses related work. Lastly, current status and future work are discussed in Sections 8 and 9, respectively, followed by the conclusion.

2 SOFTWARE DEPLOYMENT LIFE CYCLE

In the past, software deployment was largely defined as the installation of a software system; a view of software deployment that is simplistic and incomplete. Software deployment is actually a collection of interrelated activities that form the software deployment life cycle. This life cycle, as defined by this research, is an evolving collection of processes that include release, retire, install, activate, deactivate, reconfigure, update, adapt, and remove.

The processes of the software deployment life cycle are performed on either the software producer or consumer side; the processes for each side are described below.

Producer-side Processes

The producer-side of the life cycle consists of two processes, ***release*** and ***retire***. The release process is the bridge between development and deployment. It encompasses all of the activities needed to package, prepare, provide, and advertise a system for deployment to consumer sites. The release package that is created contains the physical artifacts that comprise a given software system and also a description of the deployment requirements for the software system. As modifications or updates are made to the software system, the software producer must repeat the release process to create an updated release package.

When a software producer is no longer able or willing to support a given software system, it must perform the retire process. This process withdraws support for a software system or a given configuration of a software system. The retire process is distinct from the consumer-side remove process; retiring a software system makes it unavailable for future deployment, but it does not necessarily affect consumer sites where the retired software system is currently deployed. Consumers of the software system may continue to use the software without knowing that it has been retired, but the retire process should attempt to notify current users that support for the software system is withdrawn.

Consumer-side Processes

The ***install*** process is the initial deployment activity performed by a consumer. The install process must configure and assemble all of the resources necessary to use a given software system. The install process uses the package created in the release process above. For a specific package, the install process interprets the encoded knowledge and then examines the target consumer site in order to determine how to properly configure the software system for the specific site. Once installation is complete, the deployed software system is ready for use and is ready for other deployment activities.

After a software system is installed, the ***activate*** and ***deactivate*** processes allow the consumer to actually use the software system. The activate process is responsible for making a deployed software system executable or usable. For a simple tool, activation involves establishing some form of command or click-able graphical icon for executing the tool binary. In a client/server system, for example, multiple components may need to execute in parallel. The deactivate process is the inverse of the activate process. It is responsible for shutting down any executing components of an activated software system.

Throughout the lifetime a software system is installed at a consumer site, it is not a static entity with respect to software deployment. Instead, the ***reconfigure***, ***update***, and ***adapt*** processes are responsible for changing and maintaining the deployed software system configuration. These processes may occur in any order and any number of times.

The update process modifies a previously installed software system. Update deploys a new, previously unavailable configuration of a software system. An update becomes necessary when a software producer makes a change to the description of a deployed software system. The changes to the software system's description may denote a new version, a content update, or simply a description update.

The reconfigure process, like install, also modifies a previously installed software system, but its purpose is to select a different configuration of a deployed software system from its existing description.

The purpose of the adapt process is to maintain the consistency of the currently selected configuration of a deployed software system. The adapt process must monitor changes at the consumer site and respond to those changes in order to maintain consistency in the deployed software system. Adaptation becomes necessary when a change is made to the local consumer site that affects the deployed software system. For example, when a required software system file is deleted or corrupted, the adapt process determines the affected file and replaces it.

Once a software system is no longer required at a consumer site, the ***remove*** process is performed. The remove process must undo all of the changes to the consumer site that were caused by previous deployment activities for a given software system. The remove process must pay special attention to shared resources such as data files and libraries in order to prevent dangling references to a required resource. As a result, the remove process must examine the current state of the consumer site, its dependencies, and constraints, and then remove the software system in such a way as to not violate these dependencies and constraints.

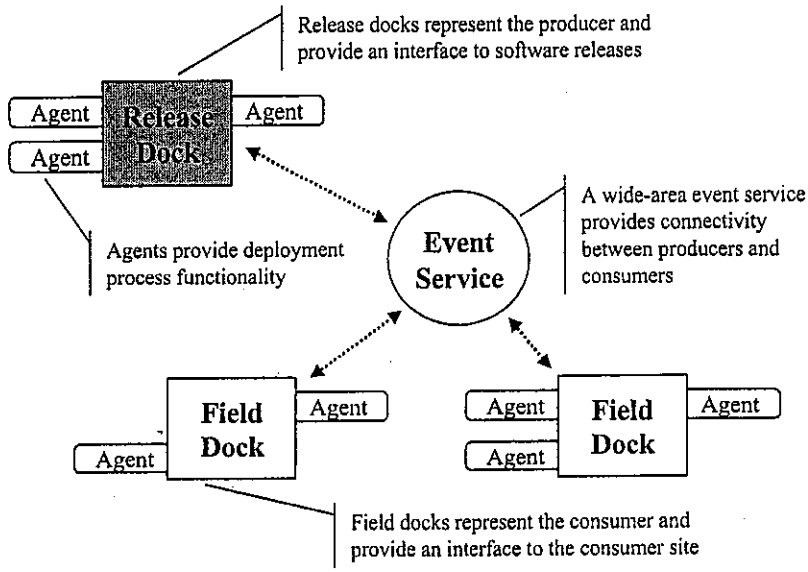


Figure 1: Software Dock Architecture

3 SOFTWARE DOCK ARCHITECTURE

The Software Dock research project, originally described in [9], addresses support for software deployment processes by creating a framework that enables cooperation among software producers themselves and between software producers and software consumers. The Software Dock architecture [see Figure 1] defines components that represent these two main participants in the software deployment problem space. The *release dock* represents the software producer and the *field dock* represents the software consumer. In addition to these components the Software Dock employs *agents* to perform specific deployment process functionality and a *wide-area event system* to provide connectivity between the release docks and the field docks.

In the Software Dock architecture, the release dock is a server residing within a software producing organization. The purpose of the release dock is to serve as a release repository for the software systems that the software producer provides. The release dock provides a Web-based release mechanism that is not wholly unlike the release mechanisms that are currently in use; it provides a browser-accessible means for software consumers to browse and select software for deployment.

The release dock, though, is more sophisticated than most current release mechanisms. Within the release dock, each software release is described using a standard deployment schema; the details of standard schema description for software systems are presented in Section 4. Each software release is accompanied with generic agents that perform software deployment processes by interpreting the description of the software release. The release dock provides a programmatic interface for agents to access its services and content. Finally, the release dock generates events as changes are made to the software releases that it manages. Agents associated with deployed software systems can subscribe for these events to receive notifications about specific release-side occurrences, such as the release of an update.

The field dock is a server residing at a software consumer site. The purpose of the field dock is to serve as an interface to the consumer site. This interface provides information about the state of the consumer site's resources and configuration; this information provides the context into which software systems from a release dock are deployed. Agents that accompany software releases "dock" themselves at the target consumer site's field dock. The interface provided by the field dock is the only interface available to an agent at the underlying consumer site. This interface includes capabilities to query and examine the resources and configuration of the consumer site; examples of each might include installed software systems and the operating system configuration.

The release dock and the field dock are very similar components. Each is a server where agents can "dock" and perform activities. Each manages a standardized, hierarchical registry of information that records the configuration or the contents of its respective sites and creates a common namespace within the framework. The registry model used in each is that of nested collections of attribute-value pairs, where the nested collections form a hierarchy. Any change to a registry generates an event that agents may receive in order to perform subsequent activities. The registry of the release dock mostly provides a list of available software releases, whereas the registry of the field dock performs the valuable role of providing access to consumer-side information.

Consumer-side information is critical in performing nearly any software deployment process. In the past, software deployment was complicated by the fact that consumer-side information was not available in any standardized fashion. The field dock registry addresses this issue by creating a detailed, standardized, hierarchical schema for describing the state of a particular consumer site. By standardizing the information available within a consumer organization, the field dock creates a common software deployment namespace for accessing consumer-side properties, such as operating system and computing platform. This information, when combined with the description of a software system, is used to perform specific software deployment processes.

Agents implement the actual software deployment process functionality. When the installation of a software system is requested on a given consumer site, initially only an agent responsible for installing the specific software system and the description of the specific software system are loaded onto the consumer site from the originating release dock. The installation agent docks at the local field dock and uses the description of the software system and the consumer site state information provided by the field dock to configure the selected software system. When the agent has configured the software system for the specific target consumer site, it requests from its release dock the precise set of artifacts that correspond to the software system configuration.

The installation agent may request other agents from its release dock to come and dock at the local field dock. These other agents are responsible for other deployment activities such as update, adapt, reconfigure, and remove. Each agent performs its associated process by interpreting the information of the software system description and the consumer site configuration.

The wide-area event service [2] in the Software Dock architecture provides a means of connectivity between software producers and consumers for "push"-style capabilities. Agents that are docked at remote field docks can subscribe for events from release docks and can then perform actions in response to those events, such as performing an update. Direct communication between agents and release docks is provided by standard protocols over the Internet. Both forms of connectivity combine to provide the software producer and consumer the opportunity to cooperate in their pursuit of software deployment process support.

4 DEPLOYABLE SOFTWARE DESCRIPTION FORMAT

In order to automate or simplify software deployment processes it is necessary to have some form of deployment knowledge about the software system being deployed. One approach to this requirement is the use of a standardized language or schema for describing a software system; this is the approach adopted by the Software Dock research project. In such a language or schema approach it is common to model software systems as collections of properties, where semantic information is mapped into standardized properties and values. This approach is also used in [4], [6], [10], [20], and [22].

Minimally five classes of semantic information have been identified [7] that must be described by the software system model. These classes of semantic information are:

- **Configuration** – describes relationships inherent in the software system, such as revisions and variants, and describes resources provided by the software system, such as deployment-related interfaces and services.
- **Assertions** – describe constraints on consumer-side properties that must be true otherwise the specific deployment process fails, such as supported hardware platforms or operating systems.
- **Dependencies** – describe constraints on consumer-side properties where a resolution is possible if the constraint is not true, such as installing dependent subsystems or reconfiguring operating system parameters.
- **Artifacts** – describe the actual physical artifacts that comprise the software system.
- **Activities** – describe any specialized activities that are outside of the purview of standard software deployment processes.

The Software Dock project has defined the Deployable Software Description (DSD) format to address these needs. The DSD is a critical piece of the Software Dock research project that enables the creation of generic deployment process definitions.

DSD provides a standard schema for describing a software system family. In this usage, a family is defined as all revisions and variants of a specific software system. The software system family was chosen as the unit of description, rather than a single revision, variant, or some combination, because it provides flexibility when specifying dependencies, enables description reuse, and provides characteristics, such as extending revision lifetime, that are necessary in component-based development.

A DSD family description is broken into multiple elements that address the five semantic classes of information described above. The sections of a DSD family description are identification, imported properties, system properties, property composition, assertions, dependencies, artifacts, interfaces, notifications, services, and activities. Some of these sections map directly onto the five semantic classes of information, others, such as system properties, property composition, interfaces, and notifications, combine to map onto the configuration class of semantic information.

A DSD family description is a simple, hierarchical schema that is built around the notion of properties of the described software system. For example, a typical property of a software system is version number. By defining such a property in a family description it is possible to organize the other pieces of the family description, such as assertions, dependencies, and artifacts, with respect to a given

version number. Other examples of software system properties are performance variants and optional capabilities. Once the properties of a software system are defined then the property composition section is used to describe the relationships among properties. For example, one property may exclude another property or it may require secondary property selections. Therefore, composition rules describe valid configurations for the described software system.

The remaining DSD family description sections are guarded by arbitrary boolean property expressions that indicate whether a specific schema element is applicable to a specific configuration. The property expression guards are expressions over software system properties, consumer site properties, or both.

The following examples depict portions of a DSD description that describes a software system that has optional online help documentation. To describe the optional online help documentation, it is necessary to create a software system property to represent the online documentation:

```
Property {
  Name = "Online Help"
  Type = "Boolean"
  Description = "Include online help."
  ... }
```

The above property definition creates a boolean property of the software system that is used for determining whether the online help documentation is applicable to a given configuration of the software system.

Also consider that the described software system only supports the Solaris™ and Window 95™ operating systems. To guarantee that these constraints are true an assertion is created:

```
Assertion {
  Condition = "($OS$ == 'Solaris') ||
              ($OS$ == 'Win95')"
  Description = "Test for supported
                 operating system."
  ... }
```

This assertion tests the target consumer site's operating system properties by using the standard namespace that is created by the field dock registry. In the above assertion example, the variable OS is actually shorthand introduced for brevity; the actual variable is the standard field dock registry path expression of:

```
$/Local/Software/OperatingSystem/Name$.
```

The artifacts that comprise the online help documentation must also be described:

```
Artifacts {
  Guard = "($Online Help$ == true)"
  Artifact {
    Guard = "($OS$ == 'Solaris')"
    SourceName = "help.html"
    Source = "/proj/doc"
    DestinationName = "help.html"
    Destination = "doc"
    Mutable = false
    Signature = "a4ca443b8902d3410ec832"
    Type = "DOCUMENTATION"
    ... }
  Artifact {
    Guard = "($OS$ == 'Win95')"
    SourceName = "help.hlp"
    Source = "/proj/doc"
    DestinationName = "help.hlp"
    Destination = "doc"
    Mutable = false
    Signature = "9283cd2378102f1a3b12ee"
    Type = "DOCUMENTATION"
    ... } }
```

The artifacts are described by nesting them in an artifact collection. The above artifact collection is guarded by a property expression that tests the applicability of the artifact collection with respect to a specific configuration; in this case, the artifact collection is only applicable if the "Online Help" property of the software system is true. The actual online help documentation artifacts are described within the artifact collection, each of which is guarded by a property expression that tests for a specific consumer site operating system value. The end result is that the proper artifact is installed with respect to the target consumer site and the selected configuration of the software system.

As a note, software system properties are arbitrary names; they have no meaning within DSD. Therefore, a property such as "version" has no special significance in DSD as it might in other configuration management disciplines. One result of this approach is that properties can be used to organize a software system in a variety of ways. For some examples, properties can be mapped to the traditional configuration management view of versions, the components in the software system architecture, or the features or capabilities of the software system.

5 SOFTWARE DOCK PROCESSES

In the prototype Software Dock framework, agents define the software deployment processes. In general, the other components in the Software Dock architecture are passive elements, such as data and interfaces. Agents, on the other hand, are active since they perform the functionality of the software deployment life cycle processes. The Software Dock framework enables the creation of a collection of generic agents that perform many of the standard software deployment processes, such as install, update, adapt, reconfigure, and remove. These generic agents, although useful in many cases, may not be sufficient for every case and therefore are also useful as base classes for the creation of other, more specialized deployment agents.

All agents perform their deployment processes by encoding some functionality that is then parameterized by the information provided in the DSD specifications and the con-

sumer site descriptions. In this fashion, a single agent definition is used for any software system described using DSD and at any consumer site that has a field dock. The remainder of this section describes the generic deployment process algorithm that all current deployment agents perform and then describes each specific deployment process in more detail.

Generic Deployment Process Definition

As described in Section 4, DSD models a software system based on properties and the proper configuration of those properties. A result of this approach led to the discovery of an abstract deployment process algorithm.

Most software deployment processes can be characterized as the transformation of one software system configuration to another based on the set of property values for a given software system configuration. A valid set of software system property values represents a particular valid configuration of a software system. Given a new set of valid property values, a deployment process simply transforms its current configuration to the new configuration by performing differential processing over the applicable schema elements of the DSD specification. The applicable schema elements for a software release are computable via the guard conditions that are dispersed throughout the DSD specification. Differential processing of the applicable schema elements creates a new software system configuration that corresponds to the desired software configuration. For a common example, if the version of a software system is changed from "1.0" to "1.1," then all of the artifacts associated with version "1.0" are removed, the artifacts associated with version "1.1" are added, and any common artifacts are left untouched.

The install, update, reconfigure, adapt, and remove software deployment processes all follow this general, abstract algorithm.

Specific Deployment Process Definitions

The software deployment processes vary from each other in small, but important ways. Each specific deployment process is described below. There is an interesting, implicit issue with respect to all of the deployment process implementations described below. All of the agents manipulate the DSD specification of a given software release in isolation of the software system itself. This means that an agent needs only the specification of a software release to perform a large portion of its tasks. As a result, an agent is much more efficient, especially in the area of transfer time, since by manipulating the schema description first, the agent only requests exactly what it needs to finish its task. This is possible since the release dock works in cooperation with the agents to perform the deployment processes.

Install Process

The install agent deploys a new configuration of a software release to a consumer site. The install agent differs from the other software deployment process agents since it is not associated with an existing software release configuration. The install agent performs its task by first retrieving the current DSD specification for the software family for which it is responsible. The install agent queries the local field dock and the user to determine the configuration of the software release to install [see Figure 2]. Once a configuration is determined the install agent only needs to perform the actions associated with all of the applicable schema elements for the selected configuration, such as testing assertions, resolving dependencies, and retrieving artifacts. Once the install process is complete, the install agent is no longer needed and therefore it removes itself. Multiple install requests are always handled by separate install agents and therefore always install another configuration of the associated software release. If a software release is unable to have multiple installations at a site, it is necessary to add an assertion to the DSD specification that tests for this condition. Currently the install process is always invoked either directly or indirectly by a specific user request to install a software release; therefore the install process is always "pull" oriented.

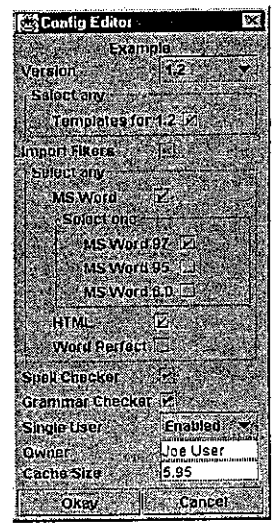


Figure 2: Configuration Editor

Update Process

The update agent deploys a new, previously unavailable configuration of a deployed software release, thus eliminating the previously deployed configuration. The newly available software release configuration is provided in an updated DSD specification for the software release. The update agent must retrieve the new DSD specification from its release dock in order to perform the update. The update agent must account for the existing deployed software release by performing differential processing on the applicable schema elements for the existing and updated software release configurations. Differential processing requires the undoing of schema elements corresponding to the prior configuration and performing the associated actions of the schema elements for the updated configuration. Any schema elements that are shared among configurations are left untouched. A specific update agent always handles the update process for a specific deployed software release. The update process is either specifically directed by the "push" of a new configuration, such as a new version, or it may be undirected in the case of a "pull" update where a new configuration is discovered or specifically selected by the user. An update is not always the result of a change to the currently selected configuration; a content-only update is also possible. In such a scenario, the update does not change the selected configuration of the software system,

only the content of the current configuration. This is typical in many software systems that use a "channel" or content delivery model. Finally, an update may not actually update the deployed software release at all; an update may simply provide a new, more accurate DSD specification for the deployed software release.

Reconfigure Process

The reconfigure agent changes the current configuration of a deployed software release, thus eliminating the previously deployed configuration. The reconfigure agent differs from the update agent because it does not retrieve a new DSD specification from its release dock even if one exists; therefore the reconfiguration agent cannot perform an update. The reconfigure agent only manipulates the existing DSD specification of the deployed software release with which it is associated. The reconfigure agent determines the new software release configuration much like the install agent. Once a new configuration is chosen from the existing DSD specification, the reconfigure agent performs differential processing on the applicable schema elements much like the update agent. A specific reconfigure agent always handles the reconfigure process for a specific deployed software release. Currently the reconfigure agent operates in "pull" mode.

Adapt Process

The adapt agent maintains the consistency of a currently deployed software release configuration in the context of the consumer site. The adapt agent does not change the software release configuration at all, it enforces it. When invoked, the adapt agent uses the existing DSD specification for its associated software release to verify that the deployed software release matches its description. It does this by determining the applicable schema elements for the deployed software release configuration and then testing them to make sure that they are still valid. If any discrepancies are discovered, the adapt agent simply performs the default processing of the invalid schema elements in order to correct the problem. A specific adapt agent always handles the adapt process for a specific deployed software release. Currently the adapt agent operates in "pull" mode. The adapt agent is easily extended to operate in "push" mode where consumer-side events, such as file deletions, automatically instigate the adapt process.

Remove Process

The remove agent is responsible for removing a deployed configuration from a consumer site. The remove agent must ensure that no constraints are violated by the removal of the software system. For example, if other deployed software systems depend on the software system that is being removed, the remove must fail. A specific remove agent always handles the remove process for a specific deployed software release. One remove request may cause multiple remove requests to other remove agents in the case of dependent software releases. Currently the remove agent operates in "pull" mode.

6 SECURITY

Security has an impact on the Software Dock research, but has not been a primary research issue. Despite this fact, this issue has not been summarily excluded in the solution discussed thus far.

Mobile agents cause a large security concern because they come from unknown sources. In order to address some of the security concerns in the Software Dock, agents operate in the Java Virtual Machine (JVM) sandbox. The field dock is the only local interface that an agent has to perform its tasks. To extend the interface provided to agents, the field dock uses a capability approach. The capability approach provided by the field dock allows access to certain restricted operations, such as controlled access to the disk. Currently, the JVM does not support a true capability approach, but this functionality is expected in the 2.0 release of Java. Regardless, all current agents are implemented as though this approach was in effect; thus there is a relatively simple transition when support for the capability-based security approach is released. In addition, this approach can be extended to adopt a mechanism by which agents can become trusted. In such a scenario, trusted agents may be provided with even more sensitive capabilities.

7 RELATED WORK

Software deployment intersects a number of related technologies; this section only covers the most important of these. For more detailed information on related technologies refer to [3] and [8].

The DSD schema created for the Software Dock project is not a unique attempt to create a standard schema for describing software systems. A handful of related technologies are also trying to address the same issue with similar approaches. Traditional configuration management modeling approaches, such as Adele [6] and PCL [22], have influenced DSD, particularly in the area of configuration selection. These traditional approaches, though, are more general configuration modeling languages that do not address software deployment. Nor do these approaches attempt to create a standard schema for any specific task, rather the modeling language is their primary contribution.

A recent, high-profile effort to create a standard software deployment schema is called the Open Software Description (OSD) format [10]. This effort is a collaboration between Microsoft and Marimba to create a schema for describing software systems for "push" technologies. OSD is immature and merely allows for the description of multiple coarse-grain variants of a single revision of a software system; dependent software systems may also be specified. The descriptive information includes some identification information and pointers to archives where the physical artifacts are found. The resulting description is too simplistic to perform any significant software deployment process automation.

The Desktop Management Task Force (DMTF) has created the Management Information Format (MIF) [4]. It is a

modeling language for describing various computing system elements. DMTF formed a specific working group to create a standard schema in MIF for describing software systems [4]. An extension to the Software MIF was created by Tivoli and is called the Application Management Specification (AMS) [20]. Since AMS is a superset of MIF, only AMS is discussed here. AMS is more mature than OSD. AMS describes a single revision of a single variant of a software system in great detail. Software system composition, constraints, dependencies, identification, support, and artifacts are some of the elements that AMS describes. AMS is not intended, though, to automate all of the software deployment processes. Instead, AMS describes a semi-static configuration of a software system that is to be installed and monitored at a consumer site; the notion of manipulating internal software system properties like revisions or variants is not directly supported. It is also assumed that there is no cooperation between software producers and software consumers; rather, there is a centralized "administration" authority that is responsible for maintaining the state of deployed software systems.

The Defense Information Infrastructure Common Operating Environment (DII COE) [13] is a Department of Defense effort to restrict the set of components used to build their software systems. The COE supports, among other things, a standard means for packaging components for delivery and installation. These packages are called *segments* [14], where each segment is a separate, installable entity. The DII COE segment describes the constraints, dependencies, and artifacts of a software system. High-level software deployment process support is provided in the form of scripts, though all deployment activities are not directly supported. Like other approaches, the deployed software system configurations are largely static entities that do not change and cannot be manipulated. The support provided is intended for a centralized administration authority and there is no release-side support.

Other approaches, such as GNU Autoconf [16], try to resolve consumer site description by using scripts and heuristics to directly examine the state of a site, but these methods are not always accurate and they are not rich enough to support deployment process automation. The Microsoft Registry [11], is a hierarchical registry of consumer site information for the Windows platform. The schema used in this registry is only partially standardized and even the standardized portions are not sufficient to semantically describe software systems for deployment.

The Redhat Package Manager (RPM) [1] is a tool for the Linux user community that provides many software deployment features. RPM packages contain the software system to be deployed and a semantic description of the software system; this description includes constraints, dependencies, artifacts, and activities in the form of scripts. The granularity of an RPM package is a single revision and a single variant. As a result, only limited forms of configuration selection are supported. RPM does not have a notion of a "release-side" and therefore is only able to request and manipulate complete packages. Also, RPM is intended for single-site deployment and provides no support for multi-site deployment or management.

A host of install utilities exist in the commercial world, such as InstallShield [12]. These systems typically work well for installation, but only address a handful of deployment processes, such as reconfigure and remove, in a limited form. Recent install utilities, such as netDeploy [19] and PC-Install with Internet Extensions [23], are starting to leverage the connectivity of the Internet. Some of these utilities are addressing the update process as well. In general, most of these solutions do not provide reasonable software system description capabilities. The deployment information is not declarative and is not rich enough for software deployment process automation.

Another class of commercial and research utilities exist to support artifact update; these systems include Castanet [17], NSBD [15], and rsync [21]. In most of these systems, there is little if any support for other software deployment processes. These solutions provide only a very simple model for describing software systems, in most cases a software system is merely considered to be a collection of files.

8 CURRENT STATUS

A prototype of the Software Dock deployment framework exists. The Software Dock prototype is implemented entirely in Java and uses Voyager [18] from ObjectSpace as an inter-process communication mechanism and a mobile agent enabling technology. A related research project at the University of Colorado, called SIENA [2], provides a wide-area event.

An evolving definition of the DSD also exists. The current definition of the DSD contains the main elements to support gross software deployment behavior.

The current implementation of the Software Dock infrastructure includes elements for both the release-side and the consumer-side. A release dock implementation exists to house the various software system releases that a software producer has available. The creation of release packages for the release dock is supported by a schema editing tool. This simple schema editor provides a way to create and edit DSD descriptions of software systems and automates some tasks, such as the entry of software artifacts into the DSD description. The schema editor is also used to submit new or updated software release specifications to the local release dock so that they can be made available for deployment. The submission of a release to the release dock automatically generates a set of HTML pages for the new release that consumers can browse and use to initiate installation.

The consumer-side the field dock describes various aspects of the consumer site, such as platform, operating system, memory, and resources. The field dock also provides a

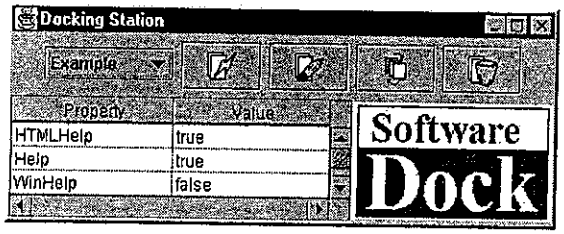


Figure 3: Docking Station Support Tool

place for agents to "dock" and perform software deployment related tasks by providing an interface to the underlying consumer site. To further support the consumer-side, a tool, called a docking station [see Figure 3], has been created that provides an interface to the software systems that have been deployed at the consumer site. The docking station provides an interface to the deployment processes that can be performed on the locally deployed software systems. The docking station is used to request updates, reconfigures, adapts, and removes.

A collection of generic agents exists to interpret the DSD software system descriptions in order to perform specific software deployment processes. These generic agents include install, reconfigure, update, adapt, and remove. Each of these agents is fully parameterized by the DSD software description. All agents generically perform the configuration and selection process and then check assertions, resolve subsystem dependencies, and request and retrieve physical artifacts. The end result is support for the release and deployment of configurable content software systems.

The current implementation was used in a demonstration to describe a Web-based software system called the Online Learning Academy (OLLA) created by a division of Lockheed Martin. OLLA consists of 45 megabytes in over 1700 files. OLLA is comprised of two dependent subsystems called Disco and Harvest. The software deployment processes of release, install, reconfigure, update, adapt, and remove have all been initially demonstrated using the generic agents described in this paper along with the DSD description of all three software systems.

Experiments were also conducted to verify the feasibility of the Software Dock. These experiments compared the Software Dock prototype to an existing deployment solution for a specific software system. A DSD specification for versions 1.1.6 and 1.1.7 of the Java Development Kit (JDK) by Sun Microsystems was created in order to compare the Software Dock deployment processes to the standard InstallShield self-extracting distribution archive for the Microsoft Windows platform. Time to completion was the dimension for comparison; Table 1 summarizes the results of the experiments.

In these experiments the Software Dock prototype performed better in most cases, even though it is dynamically creating release packages for each operation. In two of the experiments, reconfigure (remove) and update, the InstallShield process is not actually performing the equivalent actions and therefore direct comparison is difficult.

9 FUTURE WORK

The current implementation of the Software Dock concentrates on the one-to-one aspects of the software producer/consumer relationship. There is no inherent limitation in the Software Dock framework for supporting other aspects of the software producer/consumer relationship. The most obvious scenario is that of the administrator role at a consumer site.

In order to support an administrator role, a new collection of "remote" agents will be created. These remote agents will behave much like the current agents, except that they will also be parameterized by consumer site names. With such a capability, an administrator is able to specify that an activity, such as install or update, should occur on a specific site or a specific set of sites.

To further support the administrator role, a new server, called the interdock, will be introduced. An interdock server contains a global view of the consumer organization, such as site domains and global services. With the interdock, administration tasks are simplified and more complicated deployment scenarios are addressable, such as those of distributed, coordinated software systems.

In addition, the DSD will continue to be extended and expanded. Support for administration policies will be enhanced. Arbitrary dependency specification, rather than just subsystem dependencies, will also be researched. Lastly, better support for specialized deployment activities will be further investigated.

10 CONCLUSIONS

Software deployment is not a single process, such as install, but rather it is a collection of interrelated processes that are performed after a software system has been developed and made available to consumers. Support for software deployment by software producers was neglected until recently. Large network environments, such as the Internet, offer connectivity that enables software producers to offer high-level software deployment services to their customers, services that were previously not possible. By combining the connectivity of large networks with the deployment technologies described in this paper, the Software Dock creates a cooperative framework that supports software deployment.

The Software Dock supports software deployment processes by introducing components that represent software

	Software Dock	InstallShield
Install	172.0s	168.0s
Remove	36.7s	80.0s
Reconfig (remove)	40.3s	90.0s
Reconfig (add)	113.3s	284.3s
Update	187.3s	149.6s

Table 1: Software Dock Comparison Experiments

producers and consumers, release docks and field docks, respectively. The definition and use of a standard schema for describing software systems is central to the Software Dock framework, and it provides, in a declarative form, all of the knowledge necessary to perform software deployment processes. Finally, agents are employed to embody the actual functionality of the deployment processes. The agents realize the deployment process functionality in a generic fashion by interpreting the declarative schema description of the software system.

ACKNOWLEDGMENTS

This material is based upon work sponsored by the Air Force Materiel Command, Rome Laboratory, and the Defense Advanced Research Projects Agency under Contract Numbers F30602-94-C-0253 and F30602-98-2-0163. The content of the information does not necessarily reflect the position or the policy of the Government and no official endorsement should be inferred.

REFERENCES

1. E. C. Bailey. "Maximum RPM," Red Hat Software, Inc., ISBN: 1-888172-78-9, Feb. 1997.
2. A. Carzaniga, D. S. Rosenblum, and A. L. Wolf. "Design of a Scalable Event Notification Service: Interface and Architecture," Technical Report, Dept. of Computer Science, University of Colorado, 1998.
3. A. Carzaniga, A. Fuggetta, R. S. Hall, A. van der Hoek, D. Heimbigner, A. L. Wolf. "A Characterization Framework for Software Deployment Technologies," Technical Report CU-CS-857-98, Dept. of Computer Science, University of Colorado, April 1998.
4. Desktop Management Task Force, "Enabling your product for manageability with MIF files," Nov. 1994.
5. Desktop Management Task Force, "Software Standard Groups Definition, Version 2.0," Mar. 27, 1996. http://www.dmtf.org/tech/apps.html.
6. J. Estublier and R. Casallas. "The Adele Configuration Manager," Configuration Management, Wiley, 1994, pp. 99-134.
7. R. S. Hall, D. Heimbigner, and A. L. Wolf. "Requirements for Software Deployment Languages and Schema," Proc. of the 1998 Int'l Workshop on Software Configuration Management, July 1998.
8. R. S. Hall, D. Heimbigner, and A. L. Wolf. "Evaluating Software Deployment Languages and Schema," Proc. of the 1998 Int'l Conf. on Software Maintenance, IEEE Computing Society, Nov. 1998.
9. R. S. Hall, D. Heimbigner, A. van der Hoek, A. L. Wolf. "An architecture for Post-Development Configuration Management in a Wide-Area Network," Proc. of the 1997 Int'l Conf. on Distributed Configurable Systems, IEEE Computing Society, May 1997, pp. 269-278.
10. A. van Hoff, H. Partovi, T. Thai. "The Open Software Description Format (OSD)," Microsoft Corp. and Marimba, Inc., 1997. http://www.w3.org/TR/NOTE-OSD.html.
11. J. Honeycutt. "Using the Windows 95 Registry," Que Publishing, Indianapolis, IN, 1996.
12. InstallShield Corp. InstallShield, 1998. http://www.installshield.com.
13. Joint Interoperability and Engineering Organization. "Defense Information Infrastructure Common Operating Environment Baseline Specificiations," Version 3.0, Defense Information Systems Agency, CM-400-25-05, Oct. 31 1996. http://spider.osfl.disa.mil/cm/baseline/base_line3/basel in3.pdf
14. Joint Interoperability and Engineering Organization. "How to Segment Guide," Version 4.0, Defense Information Systems Agency, Dec. 30 1996. http://spider.osfl.disa.mil/cm/how_to/howtoseg.pdf.
15. Lucent Technologies. Not So Bad Distribution (NSBD), 1998. http://www.bell-labs.com/project/nsbd/.
16. D. Mackenzie, R. McGrath, and N. Friedman. "Autoconf: Generating Automatic Configuration Scripts," Free Software Foundation, Inc, April 1994.
17. Marimba, Inc. "Castanet Product Family," 1998. http://www.marimba.com/datasheets/castanet-3_0-ds.html.
18. ObjectSpace, Inc. Voyager, 1998. http://www.objectspace.com.
19. Open Software Associates. OpenWEB netDeploy, 1998. http://www.osa.com.
20. Tivoli Systems. "Applications Management Specification," Version 2.0, Nov. 5 1997. http://www.tivoli.com/o_products/html/body_ams_spec.html.
21. A. Tridgell and P. Mackerras. "The rsync algorithm," Technical Report TR-CS-96-05, June 1996. http://cs.anu.edu.au/techreports/1996/index.html.
22. E. Tryggeseth, B. Gulla, R. Conradi. "Modeling Systems with Variability using the PROTEUS Configuration Language," Proceedings of the 1995 International Symposium on System Configuration Management, Springer, 1995, pp. 216-240.
23. Twenty Twenty Software. PC-Install with Internet Extensions, 1998. http://www.twenty.com.

Read the attached paper and answer the following questions. Note that the time is limited, and you should budget your time carefully. You are suggested to spend 50 minutes in reading the paper and another 50 minutes in answering the questions.

1. Please describe how their approach collects relations between events.
2. Write down the pseudo code of the algorithm for grouping related work events based on relations between events.
3. Suppose there are various ways of grouping events. How do you determine whether a grouping scheme is better than the others? Describe as specific as possible.
4. Any comments on further improving the approach proposed in the paper?

Implementation and Evaluation of an Automatic Personal Workflow Extraction Method

Akira ABETA
Department of Computer Science and
Electronics Kyushu Institute of Technology
680-4 Kawazu, Iizuka, Fukuoka, 820,Japan.
Kyushu Hitachi Maxell Ltd.
4680 Ikata, Houjou,Tagawa, Fukuoka, Japan
abeta@macross.cse.kyutech.ac.jp

Ken'ichi KAKIZAKI
Department of Computer Science and
Electronics Kyushu Institute of Technology
680-4 Kawazu, Iizuka, Fukuoka, 820,Japan.
kakizaki@cse.kyutech.ac.jp

Abstract

We report the implementation and evaluation of a method to automatically extract a workflow during a user's ordinary use of a personal information management system (PIM). Our method accumulates the target event and time of the user's operations such as registrations and references. In order to organize related work events, our method extracts relations between events by calculating frequency of a user's sequential reference using a balloon-help-function and connects related work events based on the relations. Furthermore, we improve the accuracy of our organizing method by introducing a function that offers the user the list of the events related to his target event in order to encourage the user to make sequential references to related work events. We have constructed a prototype PIM and tested it with 6 users. We confirmed that our method could make groups of related work events and display them as connected graphs with relations shown. We discuss the efficiency of sharing this extracted workflow using sample graphs.

1. Introduction

In actual enterprises, workers handle various routine work based on their experience and knowledge. In order to perform this routine work, there are many cases in which the business process and general knowledge of experienced workers can be used by other workers as well. However, most of this knowledge is only in the minds of experienced workers. There are few cases which other workers can use this knowledge effectively. If this knowledge is recorded in some way, it becomes possible to reuse at the individual and organization levels when done systematically, and to improve not only individual ability but also the productivity of the whole organization. There are some systems [1, 2] that share and reuse knowledge of work by accumulating it in a computer. However, the time required for composition and entry of text and data in these systems is large. The benefits are usually exceeded by the costs in time and effort using such systems.

Recently many enterprises have begun using groupware [3]. Comprehensive management of employee schedules using the PIM function contained in a groupware has been accomplished in some of the enterprises [4]. When workers use a PIM [5, 6] in enterprises, they input a series of work events, such as deadlines and tasks, which comprise a work project. So schedules for a work project consist of some detailed events and relations between them, such as order and timing. Therefore, we consider that schedules for a work project present a "workflow" as knowledge of work process in a PIM. We call this workflow on a PIM as "personal workflow". From this viewpoint, we are researching a system [7, 8] that shares and reuses a workflow of a work project as knowledge of work by extracting and accumulating it from their operation records automatically.

As our system records the user's reference operations, the user is usually just viewing events on his or her calendar without involving input device operations, so determining the user's target event and time of reference is difficult. In order to record the user's reference behavior, we introduced a balloon-help function (the solid ellipse in Fig.1) which provides detailed information for each event. We ran a trial prototype PIM with this function, and confirmed [8] that our method can record a user's reference actions efficiently.

In this paper, we outline our method for organizing related work events, and report the results of trials with the prototype PIM. This method represents the organized related work events using a connected graph that indicates events with nodes and relations between events with arcs, we discuss some examples extracted and represented by this method.

0-7695-0368-3/99 $10.00 © 1999 IEEE

2. Personal Workflow

2.1. Work Management on a PIM

A PIM user inputs events with memos in a calendar view as shown in Fig.1, and will refer to this information as needed. In addition, when a user manages schedules of a work project on a PIM, he or she registers a series of events related to the work project. In order to perform events efficiency, a user arranges schedules among their events.

However, an ordinary PIM just shows the individual events and doesn't manage the relations between events. As shown in Fig.1, it is impossible to grasp the relations between events unless one produced the schedule oneself, especially if a user is an inexperienced worker. In spite of inputting the information of events in a computer, there is the problem that we cannot manage that information as knowledge of a work project that includes the relations between events.

2.2. Workflow on a PIM

An experienced worker who uses the PIM imagines the relations between events shown as connecting lines in a calendar view of Fig.1, thus he or she can use the information of the PIM to manage work projects. As shown two flow charts below in Fig.1, our ideal PIM should discern the user's understanding of relations and organize related work events to produce workflows on a PIM. We call them "personal workflows". The PIM should also return the workflow so that the user can perform the work project more efficiently by using it.

2.3. Relations between events in user's actions

During confirmation of event schedules, a user's changes of eye point or instructions by mouse are performed based on relations between events as the user understands them. We believe that the user signals relations between events in the way he or her uses the PIM. We frame two hypothesis as follow.

- Hypothesis (1): User makes sequential operations to related work events frequently.
- Hypothesis (2): The sequential reference between events indicate restrictions of the execution order of these events.

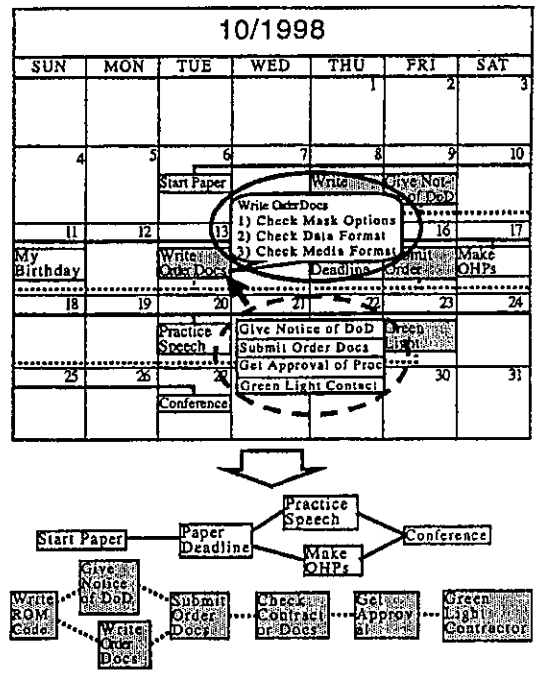


Figure 1. Relations between events on a calendar.

3. Organization related work events

Our processes for organization are : (1) extracting relations between events from the records of user's reference operations using balloon-help function, (2) intensifying the relations between events during interaction with the user by visualizing the relations and presenting them to the user, (3) making connections between events based on the extracted relations.

3.1. Extraction relations between events based on sequential references

In our ideal PIM, users can easily make sequential references to events on their calendar with the balloon-help function (as shown solid ellipse in Fig.1). Therefore, our method can get many records of a user's sequential operations. Based on our hypothesis(1), we think that users frequently make sequential references to a series of events related to a work project. A user may occasionally make only a single sequential reference, but we find the relation between events when the user frequently makes sequential references to those events. Therefore, we extract relations between two events by calculating the frequency of sequential reference operations between their events.

When the duration between two operations is shorter than a certain value, we regard the two operations to events

as a sequential operation. Then, as shown in Fig.2, the duration between two operations doesn't include duration of operations.

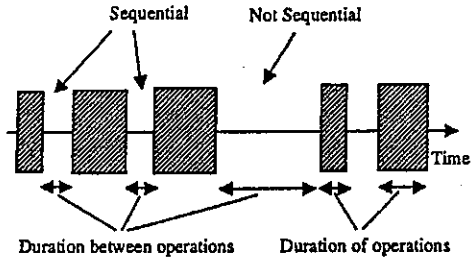


Figure 2 Detecting sequential operations.

3.2. Grouping related work events based on relations between events

We explain our method for grouping related work events by using graph theory [9] in Fig.3. It shows a time-sequential series of events registered on the PIM. In this figure, the small circles and the number of each circle indicate events and the number of each event. When a value for a relation between two events is higher than our chosen value, they are connected by an arc. So, as shown in Fig.3 (a), we get a disconnected graph, which can isolate some connected components. These connected components allow us to make groups of related work events. Furthermore, we identify which groups are work projects as distinct from individual events (such as Fig.3(b) events No.3 and No.10-11) by deleting isolated circles and connected components of circles with less than our chosen value (Fig.3(c)).

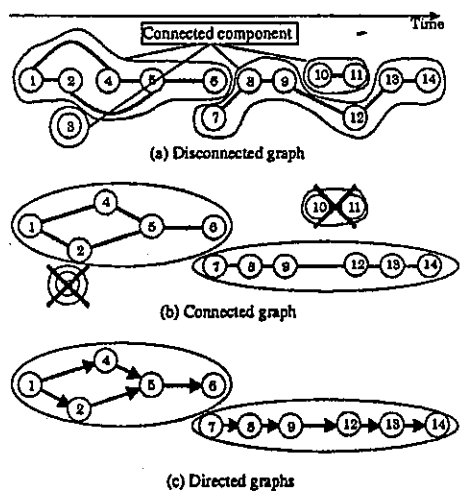


Figure 3 Process of grouping related work events.

3.3. Attaching restrictions to relations between events

In a PERT[10] used to grasp the execution order of events in project management systems [11, 12], we say that a "preceding task" must be finished before the "following task" can start, a "following task" cannot start before the "preceding task" can finish[12].

Sequential reference operations between events show time-sequential order when a user moves his or her mouse pointer. Therefore, it is considered that connected lines based on relations between events indicate the execution order of the events (Hypothesis(2)). So, the method adds restriction information to connections between events, such as "one event can't be started until another event has ended". These restrictions are indicated by arrows from preceding events to the next event. As shown in Fig.4(c), we can get some directed graphs including the order between events as workflows.

In this way, our method can organize individual events as a workflow, including relations between events, by grouping related work events and creating restrictions information as shown in Fig.4(c).

4. Intensifying the organization of work events

Users' sequential references are either:

- References to related events, or
- References to unrelated events.

There are events for work projects and individual events in a calendar. So, there are sometimes sequential references to unrelated events in records of reference extracted by analyzing records of sequential references using balloon-help. That is, groups of events extracted by our method may include unrelated events. In order to organize related work events correctly, our method needs a mechanism to sort out related work events from unrelated events. Therefore, we introduce a function that supports a user's reference only to related work events.

4.1. Related-reference-support function

When a user clicks an event on his or her calendar with reference to the target event, our PIM shows a list box for events related to the target event as shown by the broken ellipse in Fig.1. In addition, when the user selects the event out of the list, this function automatically moves the user's mouse pointer to the event that the user chose.

Using this function, the user can check a list-box which concentrates related work events that may be scattered all over the calendar. So, the user comes to make many sequential references to related work events spontaneously.

4.2. Creating the related event list

The related event list function needs to show the event that the user intends to refer to next. In addition, it is desirable that the user can easily find the event on the list. Our method assigns a value of relation between two events to indicate the frequency of sequential references to the events. So, when this value is high, a user frequently made sequential references to their events. The user would frequently make sequential references to their events in the future. Therefore, our method lists events where the value of relation against the target event is high, and indicates in order of the value in the related event list. This makes the user easily find the event to refer to from the list.

4.3. Encouraging sequential references

When the related event which the user intends to next refer to is registered in same month, the user can easily refer to the related event without turning a leaf on the user's calendar. On the other hand, when the related event is registered in another month, the user can't refer it without such operations. In additions, when the user doesn't grasp where the target event is registered, the user has to seek for it through each month. So, the user is discouraged from referencing the related work event.

Using the related-reference-support function, a user can check related work events in the list box including events registered in other months of those events. Therefore, this function encourages the user to make sequential references that the user had hesitated to make, and our method can obtain relations between events registered in other month with high accuracy.

5. Evaluation

5.1. The trials with a prototype PIM

In order to confirm the efficiency of our method, we developed a prototype PIM, and held a trial with 6 persons. In order to evaluate separately the efficiency of the balloon-help function (shown as solid circle in Fig.4) and the related-reference-support function (shown as broken circle in Fig.4), we first held a two-month trial of the prototype PIM with the balloon-help function. Next, a one-month trial, adding the related-reference-support function, was held.

The users could register, change, and refer to events on the calendar view (shown in Fig.4). Using a detail window for the information of an event, the users could input and refer to the information such as necessary resources and noteworthy points for an event. Alternatively, users could refer to the same information using our balloon-help function. Our system recorded the target events and reference times from the balloon-help function..

We implemented the extraction method in our prototype PIM. Our method extracts relations between events by finding sequential reference operations from the user's records accumulated in the database, and makes groups of related work events based on these relations. We applied our method to the user's operation records in this trial.

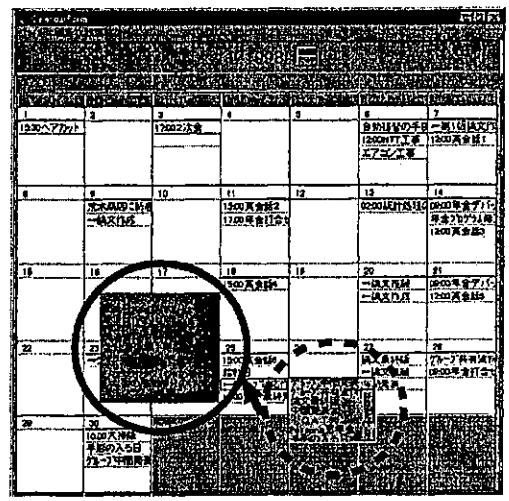


Figure 4. Calendar view of the prototype PIM.

5.2. Verification of hypothesis(1)

During this trial, for all users, we regarded two operations as being sequential when the interval between the user's two operations was less than twenty seconds. We adopted this value for the interval because it resulted in the best correspondence to the groups of truly related work events as later verified by the user.

5.2.1. Efficiency of the balloon-help function

In order to verify our hypothesis(1), we evaluate whether the users made correct sequential references to related work events. Table 1 for first two-month trial shows the number of the sequential references to events which the users stated were related or unrelated to particular work projects.

In the case of the three users A, B and C, the frequency of sequential references to work events related as the users intended was very high. But, in the case of the three users D, E and F, the frequency of sequential references to related work events was very low.

We interviewed them on this result. The first three users said that they made sequential references to related work events as they used our PIM for practical use while managing their work projects. In this case, we confirmed that our hypothesis(1) is useful. We heard from the last three users that they only used our PIM within the set time as their duty

in our trial, not for meaningful work. Our hypothesis is less useful in such cases.

Table 1 Frequency of sequential operations(1st trial).

Users	A	B	C	D	E	F
Related sequential references	293 88%	288 90%	53 76%	110 38%	45 49%	10 42%
Unrelated sequential references	39 12%	32 10%	17 24%	175 62%	47 51%	14 58%
All sequential references	332	320	70	285	92	24

5.2.2. Efficiency of related-reference-support function

Table 2 for next trial of one month shows the results of the same type evaluation as shown in Table 1. As shown in Table 2, they made about twice the frequency of sequential references compared to the first trial period. In addition, it is apparent that they made sequential references to related work events more often as well.

Therefore, we confirmed that this function encouraged users to make sequential references to related work events, and our hypothesis(1) becomes useful against all users.

Table 2 Frequency of sequential operations(2nd trial).

Users	A	B	C	D	E	F
Related sequential references	619 90%	649 91%	191 80%	425 61%	193 61%	230 69%
Unrelated sequential references	67 10%	69 9%	49 20%	269 39%	124 39%	102 31%
All sequential references	686	718	240	694	317	332

5.3. Evaluation of the method organizing related work events

5.3.1. The method for evaluation of the grouping results

We asked the users to group their events manually, and to compare and verify the groups made by them with the groups extracted by our system.

When an event is contained in our system-extracted group and also appears in the same group which the user identified manually, we say that our system correctly grouped the event (shown as solid arrows in Fig.5). If a event contained in our system-extracted group isn't contained in his group, we say that our system incorrectly grouped the event (shown as dotted arrow in Fig.5). Conversely, if the event contained in his group isn't contained in our system-extracted group, we say that our system-extracted group was short of the event (shown as broken arrow in Fig.5).

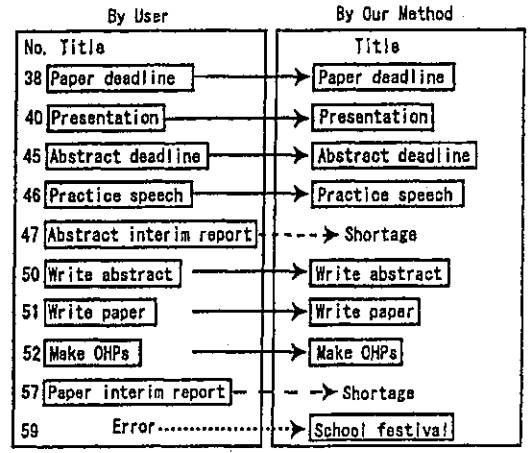


Figure 5 Evaluation of the grouping for a user.

Fig.6 shows a sample view when our prototype PIM connected related work events based on relations between events. This figure includes the system-extracted group in Fig.5 (shown as the broken frame in Fig.6). For all users in these trials, our method connected events when values for relations between events were higher than three.

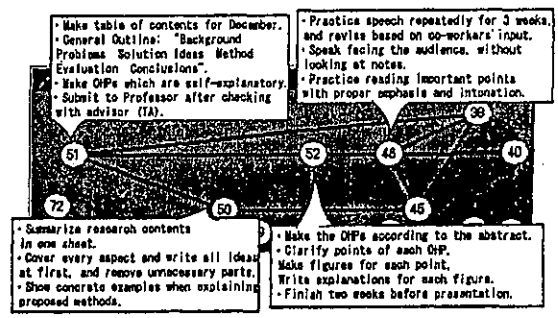


Figure 6 Sample view of disconnected graph.

5.4. Verification of Hypothesis (2)

Let us verify the hypothesis (2) by evaluating the restrictions on the execution order of events based on their relations with Fig.6. We confirmed that almost all of these lines indicated correct restrictions for the execution order of their events, such as the connections from "No.50-Write abstract" to "No.45 Abstract deadline", from "No.51 Write paper" to "No.38 Paper deadline" and from "No.46 Practical speech" to "No.40 Presentation". But some incorrect restrictions were indicated by the prototype PIM where, in fact, no restriction need exist, such as the connection from "No.51 Write paper" to "No.50 Write abstract".

We think that some of these connected lines can indicate

not only restrictions for the execution order of events but also the case where a user consults the details of events in turn in order to manage the schedule of their events. We will explore this issue in further reports.

5.4.1. The evaluation of the grouping results

Table 3 shows the grouping results for the three users A, B and C. Our method could put their events in correct groups - as high as 80 percent of the total events, and improve the accuracy ratios for these three users. Therefore, we confirmed that our method can correctly group related work events for the three users A, B and C who used the PIM for their actual work scheduling.

Table 3 Grouping results against each user.

Users	A	B	C	A,B and C
Manually grouped by user	10	5	5	20
Groups identified by user	9	4	3	16
All events	77	46	65	188
Events correct by PIM	68	37	47	152(81%)
Events incorrect by PIM	9	9	18	36(19%)
Events short by PIM	9	5	8	22(12%)

5.5. Sharing and reusing extracted workflows

With the connected graph (broken frame) in Fig.6, we discuss the efficiency with which an extracted workflow may be shared among co-workers. In Fig.6, our method made a group of seven events that a user registered to perform the work project "Graduation thesis". These were No.38: Paper deadline, No.40: Presentation, No.45: Abstract deadline, No.46: Practice speech, No.50: Write abstract, No.51: Write Paper and No.52: Make OHPs. We think that even inexperienced workers can easily make a meticulous schedule for "Graduation thesis" by reusing a workflow including all related work events and relations between their events needed to perform a " Graduation thesis".

In addition, those extracted events include details such as know-how or noteworthy points for performing those tasks. This know-how is practical knowledge gained by following the schedule, we could image that such know-how helps even inexperienced workers to successfully perform their tasks.

Generally, experienced workers (graduate students) transfer their experience to inexperienced workers (undergrads). Our method automatically extracts and accumulates such knowledge of work during a user's ordinary use on a PIM. The user can reuse accumulated knowledge of work as a typical schedule of a work project as needed. Therefore, the transfer of knowledge from experienced workers to inexperienced workers is done spontaneously, and the costs and effort of coaching the inexperienced workers can be reduced.

5.6. Refining of extracted workflows

The example of Fig. 6 showed the connected graph for "Graduation thesis" by a undergraduate student. On the other hand, Fig.7 shows one by a graduate student. In the case of the graduate student, he added No.14: Investigation of references for paper, No.15: Write idea notes and No.12: Polish paper for a "Graduation thesis", our method extracted those events together. Thus, his connected graph covers more events needed for a Graduation thesis than does the graph of the undergraduate student.

In addition, there was a qualitative difference in their knowledge of detailed information for each event. It is likely that graduate students don't make notes on points that seem obvious to them, but do make notes for particularly difficult or tricky steps which undergraduates, lacking experience, would not think of on their own.

So, our method may refine a workflow as knowledge of a work project by automatically adding necessary events and detailed information needed to perform events based on many cases of the work project [7]. We believe that our method would be able to extract and accumulate workflows which both experienced and inexperienced workers could use efficiently.

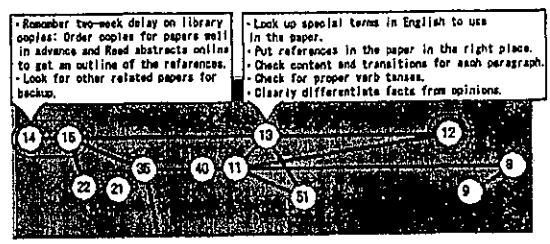


Figure 7 Sample No.2 of knowledge of work.

5.7. Presentation of Knowledge with Relations between Events

We think it is important that our method not only groups related work events but also presents them to a user as knowledge including relations between events.

In our grouping results, there were some cases in which our method grouped events of some different work projects as a single work project. Fig.8 shows a sample of such connected graph. In fig.8 our PIM made the group including the events of two works as " Write a international conference paper" in the solid frame (1) and " English Lesson " in the solid frame (2). We interviewed the user on this result. We found that because he had his paper for the international conference corrected by his English teacher,

there were some relations between some events "Write international conference paper" and some events of " English Lesson" (as shown two arrows in Fig.8). In real work project, there sometimes are weak relations among the events of different work projects, so we simply can't separate their events.

Furthermore, the broken frame in Fig.8 was a subproject of "Write international conference paper" such he surveyed books for reference of his paper and arranged them, and introduced them at a meeting. In this case, by looking such connected lines between those events, we can find that here No.122 : "Write paper" branched off from the main project and he did No.134 : "Survey books for reference", No.136 : "Make documents of books for reference" and No.135 : "Introduce of books for reference", and understand a series of events for the subproject. Therefore, in such cases that our method grouped events of some work projects as a group of a work project, users can determine their events of two different work projects, and understand the individual relations between events of each work projects by presenting them with a connected graph including relations between events.

Thus, a user can grasp a structure of relations between events of a work project and between events of some work projects. Therefore, a user can learn how he or she should perform a work project by examining and comparing their own connected graph with an experienced worker's.

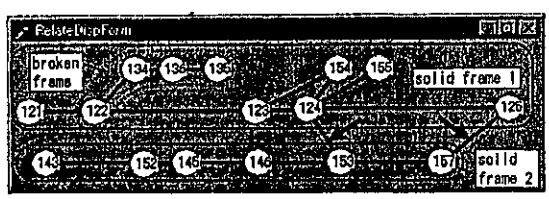


Figure 8 Sample View of Connected Graph

6. Conclusions

In this paper, we proposed that PIM users make sequential operations to related work events and we propose a method that organizes related work events based on relations between events extracted by calculating frequency of sequential operations to events. In addition, we propose a simple and efficient method for recording users' reference operations using a balloon-help-based function. We produced the prototype PIM and implemented our extraction method. We confirmed that our method can automatically group related work events by analyzing sequential references from record of user's operations when they intend to manage their work projects on PIMs. Furthermore, we introduced a related events list function that encourages users to make sequential references to related work events. We confirmed that our method can improve probability of grouping related work events even when they don't intend to manage their work projects on PIMs.

Hereafter, we will develop our proposed system to accumulate the extracted workflows using a database and to share and reuse those workflows with co-workers efficiently.

References

[1] Seki, Y., Yamakami, T. and Shimizu, A. FISH: Flexible Information Sharing and Handling System. *Trans. IEICE*, Vol.J76-D-2, No.6, pp.1223-1231, 1993, (in Japanese).

[2] Nakayama, Y., Manabe, T. and Takebayashi, Y.: Development of Knowledge/Information Sharing System, *Trans. IPSJ*, Vol.39, No.7, pp.1186-1194 (1998), (in Japanese).

[3] Lotus Development Corporation: *Lotus Notes R4.6*, (1997).

[4] Grudin, J. and Palen, L.: Why Groupware Succeeds: Dscretion or Mandate?, *Proc. 4th European Conference on Computer-Supported Cooperative Work (ECSCW'95)*, pp.263-278, (1995)

[5] Lotus Development Corporation: *Organizer Users Guide*, (1995).

[6] Now Software, Inc: Now Up To Date -- Reference Manual, Version2.1. (1991).

[7] Abeta, A., Matsunami, M. and Kakizaki, K.: A CSCW System to Aim at Sharing and Reusing the Schedule Information, *Proc. IPSJ Tech.Rep.*, 95-GW-13-2, pp.7-12, (in Japanese).

[8] Abeta, A. and Kakizaki, K.: Operation Record based Work Events Grouping Method for Personal Information Management System, *Proc. COMPSAC98*, IEEE, pp. 548-555 (1998).

[9] Ore, Oystein. and Wilson, R.: Graphs And Their, The Mathematical Association of America, (1990).

[10] Sekine,T.: *PERT-CPM*, Nikkagiren (1965). (in Japanese)

[11] Microsoft Corporation: Microsoft Project for Windows95 Users Guide (1995).

[12] Mitsubishi Corporation:M1 for Windows(1994), (in Japanese).

國立中山大學九十一學年度博士班招生考試試題

科目：資訊管理論文評述第一節【資管系】

1. Please read the attached article, "電子商務系統成功之研究模式"

   and answer the following questions:

   (a) Please provide a brief English summary (about 1000 words) of the paper. The summary should include the background and purpose of the research, the reference model and the proposed new model. (15%)

   (b) Please comment on the research and describe whether you agree on the proposed model for measuring the success of e-commerce systems and provide the reason why you agree or disagree. (15%)

   (c) If you are to do the same research, what model would you develop? (This question may be answered in Chinese). (20%)

電子商務系統成功之研究模式－Delone and MaClean 資訊系統成功模式之擴充與重新定義

王怡舜
國立彰化師範大學資訊管理系
yswang@cc.ncue.edu.tw

賴榮裕
聯華電子股份有限公司
roylai@cm1.hinet.net

摘要

本研究的主要目的是要提出一個「電子商務系統成功模式」，以作為研究電子商務系統成功的概念性架構，這模式架構可以讓不同的研究成果之間具有一個相互比較的基礎，更可以使理論的發展更為穩定且健全。首先，本研究將分析傳統「資訊系統成功模式」在電子商務環境的適用性，並探討電子商務系統與傳統資訊系統在特性上的差異，隨後本文將針對 Delone 與 MaClean 及的模式進行擴充，並提出一個探索性的「電子商務系統成功模式」以作為後續研究的概念性架構。第二、本研究將針對探索性「電子商務系統成功模式」中的變數進行概念性定義，以作為變數測量模式之基礎，並建議運用探索性的結構方程模式方法(TETRAD 方法)來檢驗先前所提出的「電子商務系統成功模式」，並對模式進行調整。第三、本研究建議後續研究可以進一步運用驗證性的結構方程模式方法(LISREL 方法)，來檢定模式的穩定性，並使本研究所提出的「電子商務系統成功模式」更具一般化。本研究所提出的模式不僅可以加速因果關係的發現及新理論的發展，且可以有效整合探索性及驗證性研究階段。此外，本研究發展出的「電子商務系統成功模式」，也有助於實務界更了解影響電子商務系統成功的因素，並對於各項自變數與因變數之間的路徑因果關係有更明確的認識，這將可以有效改善企業電子商務的行銷策略規畫，降低電子商務系統失敗的風險。

關鍵字：電子商務、電子商務系統成功、資訊系統成功、TETRAD 方法

壹、緒論

一、研究背景與動機

由於網路化與全球化市場的來臨，企業紛紛採用電子商務系統來增加其競爭優勢。因此，之前很多研究著重在探討電子商務系統成功的促進因素(facilitators)及阻礙因素(inhibitors)(e.g., Han and Noh, 1999; Turban and Gehrke, 2000)，然而這些研究成果仍存在諸多問題；因為其缺乏一個電子商務成功的概念性架構，而且對於自變數與因變數的定義仍不明確(Molla and Licker, 2001)。在公元 2000 年開始，Benbasat、Ives、與 Piccoli 針對全球資訊管理社群(ISWorld Community)進行了一項調查研究，藉以了解目前「電子商務的十大研究議題」。該項調查的結果顯示，「電子商務系統成功」是一項重大的研究議題。因此本研究的主要目的便是提出一個「電子商務系統成功模式」，本中首先將討論傳統「資訊系統成功模式」(Delone and MaClean, 1992)擴充到「電子商務系統成功模式」的可行性，提出一個探索性的概念性模式，並針對模式中的變數進行概念性及操作性定義，接著後續研究將以探索性及驗證性的結構方程模式，來檢驗「電子商務系統成功模式」，以作為實務界及後續研究之參考。

綜觀目前電子商務系統成功模式的研究缺乏一個共同的概念性架構，而且變數的定義及測量亦不明確，這對於建立網路行銷理論將有不利的影響，這不僅使得研究的結論遭受到質疑，也使得不同的研究之間缺乏一個共同的比較基礎。是故，本研究將從文獻中歸納出一個「電子商務系

統成功模式」的概念性架構，再者，由於該模式是屬於探索性的架構，因此不能直接運用驗證性的方法(例如 LISREL)來加以驗證，必須先運用探索性的方法(例如 TETRAD)來分析，而經過探索性階段的研究之後，才能進一步運用驗證性的方法來檢驗模式，完成一個研究的週期。

二、研究範圍與限制

由於不同的電子商務模式存在著極大的差異，因此本研究將研究的範圍限定在企業對消費者(B2C)的電子商務模式，並不包括企業對企業(B2B)、或消費者對消費者(C2C)等其他的電子商務型態。而且考量「資訊型網站」與「購物型網站」在消費者行為上的差異，本研究所要探討的對象，限定為經營線上購物或消費之網站，亦即必需具備線上商品展示並能提供消費者線上付款機制以及購買的「購物型網站」才納入本研究的範圍。

貳、文獻探討

一、電子商務系統與傳統資訊系統之差異

雖然企業對消費者(B2C)的電子商務系統在本質上仍是一種資訊系統，然而其與傳統組織內的資訊系統在特性上存在著若干的差異性。傳統上，資訊系統的使用者是以組織內部的員工為主，使用者常常會參與資訊系統之開發，然而電子商務系統的使用者是以組織外部的顧客為主，其很少參與系統發展的活動。再者，電子商務系統不只提供產品與服務的相關資訊，也提供了額外的交易與顧客服務功能(Young and Benamati, 2000)，而且該種系統也允許企業在線上進行三階段的行銷活動，亦即銷售前、線上銷售、以及銷售後的活動(Schubert and Selz, 2001)。透過電子商務系統，企業及消費者可以在線上進行議價、協商、下單、訂單追蹤、付款、顧客服務等活動。而對於數位產品或服務而言，消費者亦可直接從網站上取得產品及服務，而不需要實體通路及人員的配合，例如：網路下單、音樂下載等。表 1 則列示了「傳統資訊系統」與「B2C 電子商務系統」在特性上的主要差異。

電子商務系統在本質上仍是一種資訊系統，這使我們可以藉重傳統資訊系統的理論來解釋電子商務的現象，但是由於電子商務系統與傳統資訊系統仍存在諸多差異，因此我們在運用傳統資訊系統理論時，必須考量電子商務系統所提供的額外企業功能。本研究將運用資訊系統成功的模式(Delone and MaClean, 1992)來作為電子商務系統成功模式的基礎，並將其進一步的擴充。

表 1：傳統資訊系統與 B2C 電子商務系統的差異

特性	B2C 電子商務系統	傳統資訊系統
使用者參與系統之開發		+
人機的互動	+	+
提供組織內部的資訊		+
提供產品或服務的資訊	+	
線上協商與議價	+	
提供顧客下單及追蹤功能	+	
線上付款機制	+	
售後服務與顧客支援	+	

+ 經常或一定　　很少或不曾

二、資訊系統成功模式

雖然資訊系統成功的實證研究很多，但是資訊系統成功的意義仍未獲得共識(Garrity and Sanders, 1998)，因為資訊系統成功是一個多構面的概念，其必須從多個層面來評估，諸如技術、個人、群體、組

織等(Molla and Licker, 2001)。然而很多學者認為 Delone and MaClean 有關資訊系統成功的研究是一項重要的突破，其所提出的資訊系統成功模式，也納入了許多個人及組織的構面。在該模式中(如圖 1 所示)，「系統品質」與「資訊品質」會單獨或同時影響「資訊系統使用」以及「使用者滿意度」；此外，「資訊系統使用」與「使用者滿意度」會相互影響，而「資訊系統使用」與「使用者滿意度」兩個構念又會影響某些「個人績效變數」，最後「個人績效變數」對某些「組織績效變數」造成影響(Delone and MaClean, 1992)。

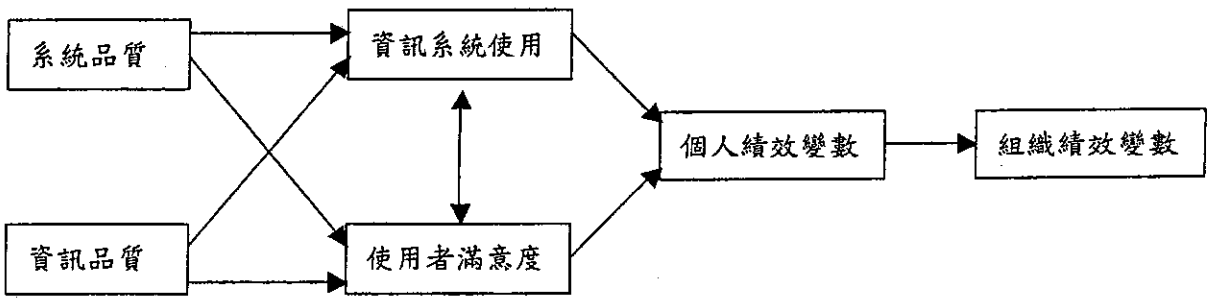


圖 1：Delone 與 MaClean 的資訊系統成功模式(資料來源：Delone and MaClean, 1992)

雖然 Delone 與 MaClean 的模式引起了很多的討論及質疑(Saarinen, 1996; Seddon and Kiew, 1996; Grover et al, 1996; Garrity and Sanders, 1998; Seddon, 1997)，但是該模式卻對資訊系統成功的變數因果關係提出了一個明確的架構，況且 Seddon 與 Kiew(1996)也對於該模式左手邊的部份提出了實證上的證據。由於，電子商務系統在本質上亦是一種資訊系統，因此 Delone 與 MaClean 可以用來作為電子商務系統成功模式的基礎，然而承如先前所述，運用傳統資訊系統成功模式到電子商務領域時，不得不考量電子商務系統在「使用者」及「系統功能」上與傳統資訊系統所存在的差異性。

參、電子商務系統成功模式

一、探索性的電子商務系統成功模式

目前有很多討論電子商務成功的研究，有些研究著重於電子商務的系統層面(e.g., Von Dran et al, 1999; Kim, 1999; Loiacono and Taylor, 1999)，而有些研究則是著重於電子商務系統的內容品質(e.g., Zhang et al, 2000)，也有些研究是著重於評估電子商務系統的可用性(usability) (e.g., Henneman, 1999; Nielsen, 1999)，最近的研究則是著重於評估電子商務顧客的滿意度(Wang et al, 2001; Han and Noh, 1999; Jones and Kayworth, 1999)、網站的服務品質(Barnes and Vidgen, 2001)、或是電子商務系統的整體績效(Schubert and Selz, 2001)。

雖然上述的研究都對於電子商務成功的衡量作出了重要的貢獻，然而大多數的研究卻沒提出一個整合的概念性模式或架構，於是 Molla and Licker(2001)提出一個概念性的電子商務系統成功模式，來說明不同的自變數與因變數之間的因果關係，然而該模式忽略了若干重要的績效變數(例如：顧客績效與組織績效)，且忽略了價格因素對於電子商務系統成功的重要性。因此本研究以 Delone 與 MaClean 及 Molla and Licker(2001)的模式為基礎，並

考量電子商務系統的特性，將其修改並擴充為一個「探索性的電子商務系統成功模式」(如圖 2 所示)，以作為後續實證研究的概念性模式與架構，並促進電子商務理論的發展。下文將說明該模式的發展過程。

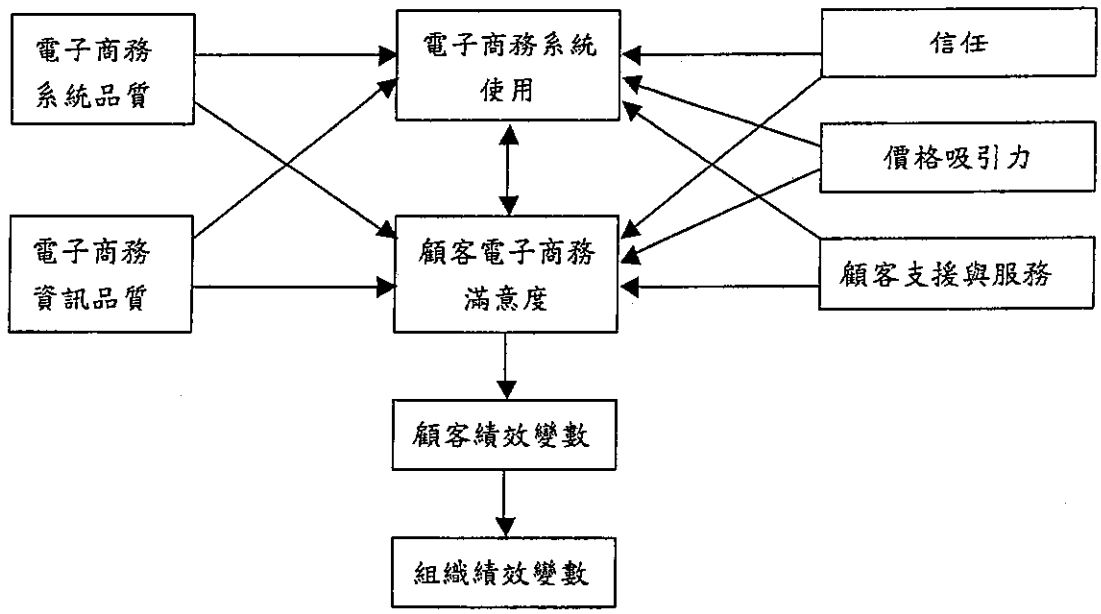


圖 2：探索性的電子商務系統成功模式

(修改自 Delone and MaClean, 1992; Molla and Licker, 2001)

由於電子商務系統在本質上仍屬於資訊系統，因此 Delone 與 MaClean 的模式中所提出的「系統品質」及「資訊品質」兩個構念，仍會影響「使用者滿意度」及「資訊系統使用」。為了區別電子商務模式與傳統資訊系統模式，本研究依循 Molla 與 Licker 的模式，將 Delone 與 MaClean 模式中的「系統品質」、「資訊品質」與「資訊系統使用」分別更名為「電子商務系統品質」、「電子商務資訊品質」以及「電子商務系統使用」；承如先前所述，電子商務系統的使用者主要是公司的顧客而非公司的員工，因此 Delone 與 MaClean 模式中的「使用者滿意度」被更名為「顧客電子商務滿意度」。另外，「個人績效變數」亦更改為「顧客績效變數」。傳統資訊系統研究中，常常將「使用者滿意度」視為是資訊系統成功的代理變數(e.g., Bailey and Pearson 1983; Ives, et al. 1983; Doll and Torkzadeh 1988)，同樣地，「顧客電子商務滿意度」也可作為電子商務成功的代理變數，然而在最近的研究亦顯示，顧客滿意度會影響顧客的忠誠度及再購買行為(Wang et al., 2001; Turban et al., 2000)、組織的獲利能力與市場佔有率(Naumann 2001; Meuter et al, 2000; McColl-Kennedy and Schneider, 2000)。因此，在電子商務成功模式中，顧客滿意度也可以視為是一項中介變數，其會直接或間接影響到顧客績效變數(例如：顧客購買忠誠、顧客態度忠誠等)以及組織績效變數(例如：公司獲利力、市場佔有率等)。

承如先前所述，電子商務系統所執行的企業功能不同於傳統的資訊系統，而且在傳統資訊系統環境中，使用者可能是被迫使用資訊系統，然而在電子商務的環境中，顧客使用電子商務系統則是完全出於自願的行為。尤其對於交易型購物網站，顧客是否願意線上購物及付款，有部分取於消費者是否信任該網站(Liao and Cheung, 2001; Ferraro, 1998; Warrington et

al, 2000)；其他的研究亦顯示，顧客對網站的「信任」會影響顧客對線上購物系統的「使用」與「滿意度」(Han and Noh, 1999)，而且 Lee and Turban(2001)亦認為 B2C 電子商務成功有一部份是決定消費者對銷售者及產品的信任。因此本研究將「信任」納入探索性的電子商務成功模式之中，作為外生變數(Molla and Licker, 2001)。不同於傳統資訊系統，電子商務系統不僅提供資訊、亦執行了線上交易、付款、以及顧客支援與服務的功能，因此電子商務成功模式必須考量公司執行網路行銷活動的各個層面；承如 Turban et al(2000)所述，網路商品行銷中的「價格吸引力」將會影響「顧客的滿意度」，消費者可以很容易在網路上進行比價，尤其是當產品規格明確時，消費者通常會「使用」具有價格吸引力的網站，也可能產生較高的「滿意度」。而 Molla 與 Licker(2001)亦主張「顧客的支援與服務」將會影響「顧客電子商務滿意度」與「電子商務系統的使用」。

二、模式中構念之概念性定義

由於上述所提出的電子商務成功模式是屬於一個探索性的架構，因此必須經過實證資料的檢定，並依據實證的結果對模式作進一步的修正。然而，在進行實證分析之前，必須對於模式中的構念進行概念上的定義，以作為後續構念操作性定義及測量模式發展的基礎。以下將分別針對模式中的構念進行探討。

(一)電子商務系統品質

電子商務系統的品質是影響電子商務系統成功的一個重要因素，然而「電子商務系統品質」與「資訊品質」是兩個不同的概念。早期資訊系統的研究認為「系統的可靠性」、「系統的彈性」、「系統的正確性」、「回應時間」、以及「容易使用」是構成「資訊系統品質」的重要構面(Delone and MaClean, 1992)。然而，這些概念仍可以適用於電子商務系統的環境。最近的研究(Turban and Gherke, 2000; Han and Noh, 1999)主張「24 小時營業」、「軟體與硬體的穩定性」、「網頁下載速度」、「系統架構」、以及「視覺化的外觀」等變數亦是構成電子商務系統品質的重要層面。在本研究所提出的模式中，電子商務系統品質將影響顧客對系統的使用及滿意度，而 Fellenstein and Wood(2000)亦認為失敗的電子商務系統將會造成約 40%重覆上線者的流失。

(二)電子商務資訊品質

電子商務系統所提供的資訊品質是影響電子商務系統成功的一個重要因素，Turban and Gehrke(2000)亦認為網站的內容品質會影響網站使用者的滿意度。對於某些行銷數位產品的網站而言，其網站所提供的資訊內容便是它的商品(例如：付費電子報)，對於這類網站而言，網站的資訊品質更為重要。然而，傳統資訊品質方面的研究與概念，亦可以適用到電子商務的環境中。Zmud(1978)認為資訊的品質包括資訊的可取用性(availability)、可理解性(comprehensibility)、相關性(relevance)、有用性(usefulness)、及時性(timeliness)、可靠性(reliability)、正確性(accuracy)、一致性(consistency)、易讀性(readability)等向度。最近的研究亦顯示，電子商務系統的資訊品質包括了「資訊特性」與「資訊展示」兩大層面(Zhang et al., 2000; Von Dran et al., 1999)，是故在測量「電子商務資訊品質」測量工具時，可以參酌上述所提出的資訊品質衡量屬性，且必須與「電子商務系統品質」構念有所區別。

(三)電子商務系統使用

電子商務系統的使用亦是評估電子商務成功的指標之一，一般電子商務中常使用點閱率來代表顧客的使用程度，然而「電子商務使用」必須更明確的定義。Young and Benamati (2000)認為電子商務系統的使用可以分為「資訊」、「交易」或「顧客服務」三個目的，因此在測量顧客對電子商務系統的使用程度時，必須同時考量這三個層面的使用程度，而不能只考量其中一個層面。在本研究的成功模式中，「電子

商務系統使用」與「顧客電子商務滿意度」都是屬於因變數，其會受多個構念所影響。

(四)顧客電子商務滿意度

Cardozo 於 1965 年最先提出「顧客(消費者)滿意度」之概念，其以實徵研究探討顧客預期與實際之差距對滿意度，以及滿意度對再購意願之影響。Howard and Sheth(1969)首先將滿意度應用於消費者理論上，他們認為顧客滿意度是購買者對於其因購買某一產品而做的犧牲(如時間、金錢....)與所得到的補償是否適當的一種認知狀態。

Churchill et al. (1982)認為在概念上，顧客滿意度是一種購買與使用產品的結果，其是由購買者比較「購買時的成本效益」與「預期的成本效益」所產生的；而操作上顧客滿意類似態度，其可以透過評估顧客對於不同屬性滿意程度的加總值來決定。Westbrook and Oliver(1991)引用 Day(1984)的定義認為顧客滿意度是對於特定購買選擇所作出的一種選後評估判斷，所以其是顧客購後評估購買前預期與購買後產品實際效果所產生差距的顧客認知反應。Fornell(1992)亦指出顧客滿意是一種以整體性的購後評估。最近，Giese and Cote(2000)綜合了 30 年來的 20 個在顧客滿意度研究中所用的定義，並參酌個人訪談與焦點群體訪談的資料後，將顧客(消費者)滿意度定義為「顧客(消費者)在某一個時間點或期間(一般指消費後，研究人員可依研究問題自行定義)，對於產品取得與消費經驗等層面，所感受到不同程度的彙總性情感反應」。依據 Giese and Cote 的架構並考量電子商務系統的目的之後，本研究將顧客電子商務滿意度定義為「顧客在消費後，對於透過電子商務系統取得資訊、進行交易、獲得顧客支援等消費經驗層面所感受到不同程度的彙總性情感反應」。在操作化層次，則必須設計多個問項來測量多個顧客滿意層面。

(五)信任

許多學者皆認為，預測顧客行為的主要架構，除了顧客對整體的滿意度外，還需加入信任度（trust）做為考量(Moorman et al., 1993; Morgan and Hunt, 1994; Garbarino and Johnson, 1999)。Lee and Turban(2001)亦認為 B2C 電子商務成功有一部份是決定消費者對銷售者及產品的信任。在傳統行銷的文獻中，多數學者皆強調信任是一種「互動」的行為，與「承擔風險」有關，信任會直接導致主動合作的產生。維持協議雙方長期的合作關係，關鍵在於成員能否相互信任(Morgan and Hunt, 1994)。而 Bettencourt(1997)的研究，亦表示顧客對於企業的信任程度越高，愈願意提供意見給企業作參考，當彼此交易關係存在著信任感時，關係人會高度地重視彼此之間的關係，而會想要去對關係作承諾以持續維持雙方的關係。因此，在合作及交易關係中，商家若能維繫消費者對其信任的程度，則可降低消費者的知覺風險，進而提高消費者的再購買行為。而在線上購物的相關文獻中，亦有許多學者強調安全性是消費者選擇上網購物的主要因素之一。Warrington et al. (2000)認為顧客會擔心在網路上提供敏感性的資訊，而 Ferraro(1998)則認為只有當消費者具有一定程度的信任之後，顧客才會使用電子商務系統。是故，依據上述的觀點，我們可以推論出，在電子商務環境中，信任可以導致較高的「電子商務系統使用程度」及「顧客電子商務滿意度」。

此外，Hoffman and Novak(1996)也曾表示：安全性是影響網路購物的因素之一。Kiely(1997)的調查亦顯示，消費者不願在線上購物的原因之一為：害怕在網際空間中失去隱私權。是故，電子商務成功模式中的「信任」可以定義為，消費者對網站所感受到的「安全性」及「隱私權保護」程度(Molla and Licker, 2001)。

(六)價格吸引力

Turban et al.(2000)認為價格吸引力會

影響電子商務中的顧客滿意度，由於產品搜尋與價格比較的網站盛行(例如：BestBookBuys.com 及 compare.net)，消費者可以很容易在網路上進行比價，尤其是當產品規格明確時，消費者通常會「使用」具有價格吸引力的網站，也因此可能產生較高的「滿意度」。

(七)顧客支援與服務

對顧客的支援與服務亦是影響「顧客電子商務滿意度」及「使用」的重要因素；電子商務系統的顧客支援服務包括了「搜尋服務」、「顧客問題回應」、「訂單狀態查詢與追蹤」、「付款」、「帳戶維護」等層面(Kardaras and Karakostas, 1999; Schubert and Selz, 2001)。

(八)顧客績效變數

在傳統資訊系統領域中，「使用者績效變數」是指資訊系統使用者的生產力，但在電子商務系統環境中，「顧客績效變數」在概念上有不同的意涵，其可以視為是顧客再購買傾向(repurchase intention)以及顧客忠誠度方面的變數；顧客忠誠可以分為態度忠誠(attitudinal loyalty)與購買忠誠(purchase loyalty)。「態度忠誠」在概念上可以定義為：一般消費者對於該網站的承諾程度，而「購買忠誠」在概念上可以定義為：一般消費者重覆上該網站購買產品的意願(Chaudhuri and Holbrook, 2001)。

(九)組織績效變數

組織績效變數一般包括公司獲利力、市場佔有率等變數；「市場佔有」的在概念上可以定義為：某網站的銷售額佔該類網站所有銷售總額的百分比，而「公司獲利力」則可以用普通股權益報酬率(ROE)，或總資產報酬率(ROA)來衡量。

肆、後續研究

一、探索性的結構方程模式分析

本階段主要的目的是要運用探索性的結構方程模式，即 TETRAD 方法，來檢驗「電子商務系統成功模式」。Lee et al.(1997)的研究認為現行 MIS 的研究應該著幾個方向改進：(1)建構更豐富的因果關係；(2)增加模式表達的彈性；(3)整合潛伏變數與明確(可測量)變數；(4)結合探索性與驗證性的研究。此外，Lee et al.亦認為目前探索性的研究工具存在諸多的缺點，例如：會將未經驗證的統計假定(untested statistical assumptions)，直接帶入驗證性階段，也因此認為無母數的 TETRAD 方法適合運用在探索性階段的研究；使用 TETRAD 的目的是要抽象化統計模式的因果結構，忽略 equation 及大部分的統計假設，亦即不使用複雜的方程式及對線性模式機率分配的假設。利用方向的圖形(directed graph)來描述模式因果關係的假設，並由圖形來決定重要的統計特性，也因此 TETRAD 並不做模式的參數估計，也就是不用計算線性係數(linear coefficients)，而只使用樣本資料的相關係數(correlation)或共變數(covariance)來判斷模式的配適度，亦即從「假設的模式」中可以推導出母體的相關係數或共變數必須滿足某些等式關係，這些等式關係有時稱為過度辨視的限制式(overidentifying constraints)，TETRAD 方法可以從研究模式所假設的因果關係圖形中決定兩種限制式，即消失的偏相關限制式(vanishing partial correlation)以及消失的四變數限制式(vanishing tetrad equation)。接者 TETRAD 經由檢定樣本資料是否與這些限制式配適，來決定模式的配適程度。是故，運用 TETRAD 方法來檢驗「探索性的電子商務系統成功模式」，可以避免有母數統計方法運用在探索性階段的缺點，並增加模式表示的彈性，以促進新理論的發展，且能使探索性研究與驗證性研究能夠相互連結(Lee et al., 1997)。

二、驗證性的結構方程模式分析

(一)衡量模式之驗證

本研究階段可先運用驗證性因素分析

(CFA)來檢驗構念的衡量模式，在確保衡量模式之信度與效度之後，才能進一步運用結構方程模式(SEM)來探討多個構念之間的關係。驗證性因素分析(CFA)常用在量表驗證與調整過程；驗證性因素分析通常用來描述及估計因素模式，而因素模式中包括了一組用來解釋可觀察變數(observed variables)的潛伏變數(latent variables) (Bagozzi 1980; Bollen 1989; Jöreskog and Sörbom 1993)。如同 Byrne (1998)所述，「驗證性因素分析適合用於當研究人員對於基礎的潛伏變數結構有一定的了解時；依據理論或實證研究的知識，研究人員事先描述潛伏變數與可觀察變數之間的關係，再進一步用統計方法來檢定假設的模式結構」(p.6)。運用驗證性因素分析可以提供測量效度的證據，也可以降低在完整結構方程模式(SEM)中由於大量測量誤差所引起的混淆(Anderson 1987; Anderson and Gerbing 1988; Jöreskog 1993; Segars and Grover 1993)。研究過程中可以使用 LISREL 軟體來進行驗證性因素分析，並檢定每一構念衡量模式的信度與構念效度(Bagozzi and Phillips 1982; Bagozzi et al. 1991)，其中構念效度包括了內容效度(content validity)、收斂效度(convergent validity)、及區別效度(discriminant validity)。

(二)構念關係之驗證

在確保每個構念衡量模式的信度與效度之後，後續研究將可運用結構方程模式(SEM)來進一步驗證模式的穩定性並對模式做必要的修正。而本研究發展出的「電子商務系統成功模式」將提供後續研究人員一個共同的概念性架構，這將使電子商務理論的發展更為穩定且健全。在理想上，模式配適度(goodness-of-fit)的評估應該從各種不同的準則來進行(Byrne 1998)。因此，除了採用絕對性的配適度指標(例如：χ^2、GFI、AGFI、PGFI、以及RMSEA 等)，亦應採用比較性的配適度指標(例如：χ^2/df、CFI)，來比較不同模式之間配適度的強弱。

伍、結論

本研究以傳統資訊系統成功模式為基礎，並結合電子商務系統的特性，提出一個探索性的電子商務系統成功模式，並建議後續研究使用 TETRAD 探索性的研究工具，來進行模式的初步檢驗與調整。隨後在驗證性的研究階段，可以進一步運用 LISREL 驗證性的研究工具，來確認模式的穩定性。本研究提供學術界在研究「電子商務系統成功」的一個概念性模式，這模式可以讓不同研究成果之間具有一個相互比較的基礎，更可以使網路行銷理論發展更為穩定且健全。而這個模式著重於構念之間因果關係的描述，該模式中可以納入了潛伏變數(latent variable)與可測量變數(measured variable)之間的關係，避免模式完全使用潛伏變數或是完全使用可測量變數的缺點，這也使得模式更為豐富化。此外，本研究採納 Lee et al.(1997)的看法，建議在模式探索階段使用 TETRAD 方法，且在驗證性階段使用 LISREL 方法，如此可以避免將未經驗證的統計假定(untested statistical assumptions)，直接帶入驗證性階段的缺點，而且運用 TETRAD 方法可以使模式的表達更具彈性。由於 TETRAD 方法的使用，將可以加速因果關係的發現及新理論的發展，且可以有效整合探索性及驗證性研究階段(Lee et al., 1997)。此外，本研究發展出的「電子商務系統成功模式」也有助於實務界更了解影響電子商務系統成功的因素，並對於各項自變數與因變數之間的路徑因果關係有更明確的認識，這將可以有效改善企業電子商務的行銷策略規畫，降低電子商務系統失敗的風險。

參考文獻

限於篇幅，讀者若需參考文獻請與作者聯絡。

國立中山大學九十一學年度博士班招生考試試題

科目：資訊管理論文評述第二節【資管系】

請閱讀所附論文：Applying the Technology Acceptance Model and Flow Theory to Online Consumer Behavior，並回答下列各問題。
(注意：各問題有標註答題字數限制，請仔細構思您的回答在字數限制內清楚完整表達(超過字數限制將扣分)。

1. 所附論文共分為五節，請簡述各節的主旨(請用中文在五百字內回答)(10%)

2. 請就論文的內容，用中文回答下列問題：
   (1) 研究結果對於假說 1a. Shopping enjoyment is positively related to intention to return.是被支持還是不被支持？作者根據怎樣的驗證程序來下結論。(5%)
   (2) 研究結果對於假說 3a. Consumers with higher shopping enjoyment are more likely to make unplanned purchase.是被支持還是不被支持？作者根據怎樣的驗證程序來下結論。(5%)

3. 請就論文的研究結果，用中文回答下列問題：
   (1) 研究結果的發現，是否增進對網路客戶行為的了解？請從研究的發現來比較網路客戶和傳統購物者的異同。 (請用中文在二百字內回答) (10%)
   (2) 您如何解釋為何各項變因與非計畫性購買(unplanned purchases)沒有顯著性的關係。如果要進一步驗證這樣的關係，亦請您提出可能的研究改進方法。(請用中文在二百字內回答) (10%)
   (3) 論文的研究結果，假設您是網路購物網站的經營者，您要如何來應用這些研究發現，請提出您的應用構想(請用中文在二百字內回答) (10%)

Applying the Technology Acceptance Model and Flow Theory to Online Consumer Behavior

Marios Koufaris
Zicklin School of Business, Baruch College, CUNY, 55 Lexington Avenue, Box B11–220, New York, New York 10010
marios_koufaris@baruch.cuny.edu

In this study, we consider the online consumer as both a shopper and a computer user. We test constructs from information systems (Technology Acceptance Model), marketing (Consumer Behavior), and psychology (Flow and Environmental Psychology) in an integrated theoretical framework of online consumer behavior. Specifically, we examine how emotional and cognitive responses to visiting a Web-based store for the first time can influence online consumers' intention to return and their likelihood to make unplanned purchases. The instrumentation shows reasonably good measurement properties and the constructs are validated as a nomological network.

A questionnaire-based empirical study is used to test this nomological network. Results confirm the double identity of the online consumer as a shopper and a computer user because both shopping enjoyment and perceived usefulness of the site strongly predict intention to return. Our results on unplanned purchases are not conclusive. We also test some individual and Web site factors that can affect the consumer's emotional and cognitive responses. Product involvement, Web skills, challenges, and use of value-added search mechanisms all have a significant impact on the Web consumer. The study provides a more rounded, albeit partial, view of the online consumer and is a significant step towards a better understanding of consumer behavior on the Web. The validated metrics should be of use to researchers and practitioners alike.

(TAM; Flow Theory; Nomological Validity; Web Skills; Value-Added Search Mechanisms; Online Consumer Behavior)

1. Introduction

Consumer behavior on the Web has been the subject of considerable research in the last few years, but understanding it is made difficult by the fact that the main entities involved, consumers and businesses, have been transformed. First, every consumer is now also a computer user. The online consumer performs all the functions of a traditional consumer on a computer while interacting with a system, i.e., a commercial Web site. S/he, therefore, also exhibits all the characteristics of a computer user. Second, the physical store has been transformed into a virtual store through information technology (IT). In the physical commercial world, the IT used for operations remains mostly in the background, invisible to the consumer. In e-commerce, however, the technology has been moved to the foreground and has become the store itself as a Web site.

Such Web-based stores that use "networks and Internet technology for communications and transactions between various groups of stakeholders like businesses and consumers" have recently also been called

1047-7047/02/1302/0205$05.00
1526-5536 electronic ISSN

Net-enabled organizations (NEOs) (Straub and Watson 2001). There is a need to understand and measure such online consumer behavior and much of that burden falls on the IS field (Straub and Watson 2001). To meet this goal, a multidisciplinary approach is ideal, and for this reason we take that approach in this study. In this study, we concentrate on new customer retention and unplanned purchases in business to consumer (B2C) electronic commerce. We also examine how certain emotional and cognitive responses to an initial Web store visit influence these two variables. In addition, we study the effects of some individual consumer differences as well as the impact of using different types of search mechanisms on the Web site. While models and constructs from marketing and psychology are prominent in this work, the research is relevant to the information systems field due to its inclusion of traditional IS variables (Straub and Watson 2001), primarily the variables of the Technology Acceptance Model (Davis 1989).

2. Theoretical Framework and Study Measures

Discussion of metrics will accompany the building of theory-based hypotheses so that we can later examine a nomological network to test construct validity. Although this approach does not follow the conventional article format, we felt that it was justifiable given the stress on metrics in the special issue.

There have been various attempts at building models of consumer behavior online. Examples include looking at "conversion" of Web surfers to Web customers (Berthon et al. 1996), studying consumer information acquisition and purchase decision online (Alba et al. 1997), and determining reasons for not shopping online (Peterson et al. 1997). Transaction cost theory has been used to explain the impact of IT on market structure (Malone et al. 1987) but also to understand individual consumer behavior. For example, buyer search costs are substantial transaction costs and they can determine buyer behavior and eventually market structures (Bakos 1997).

An important question is whether online consumers think and act differently than their offline counterparts. If so, which metrics can we use to capture these differences? For example, online consumers cannot depend on all five senses to make purchases; instead, they must rely on limited product representations such as photographs and text descriptions. Yet, studies have shown that the representation of the product online, indeed the overall quality of the shopping experience, matters for both attitude towards shopping online as well as intention to buy (Burke et al. 1992, Jarvenpaa and Todd 1997a and b, Nowlis and McCabe 2000, Novak et al. 2000). NEOs, however, cannot rely on enticing product displays, exciting music, and "hip" clientele to convince their customers to buy. Instead, they may depend on well designed Web pages and powerful Web features, such as recommender systems and one-click checkouts. The online consumer may also have a different social and work environment than the offline consumer. For example, intention to buy online is influenced by the level of the consumer's lack of leisure time, as in offline shopping, but also by the level of their wired lifestyle (Bellman et al. 1999).

A key difference between online and offline consumer behavior is that the online consumer is generally more powerful, demanding, and utilitarian in her shopping expeditions. As a result, customer loyalty on the Web is low overall (Morrisette et al. 1999); although, as in the physical world, it is influenced by the availability of good, relevant content in an enjoyable context (Rice 1997, Eighmey 1997, Eighmey and McCord 1998). The locus of power seems to be shifting from the vendor to the consumer (Raman 1997) who does not favor traditional advertising and promotions online (Maignan and Lukas 1997). Research has also shown that certain site features, such as the availability of a FAQ section or promotions at the Web store entrance, can influence both traffic on the Web site and overall sales (Lohse and Spiller 1998a and b).

Unlike offline consumers, online consumers are concerned with those risks inherent in buying on the Web, risks such as credit card fraud and not receiving the right products (Bhatnagar et al. 2000). Perceived risk of online shopping and perceived ease of use of the Web site have been shown to influence attitude towards online purchasing (Heijden et al. 2001). The impact of perceived ease of use, however, seems to vary depending on the type of task the consumer is undertaking. Its effect is more significant when consumers

are using a Web site to inquire about products rather than to purchase them (Gefen and Straub 2000).

Online consumers clearly share some characteristics of their offline counterparts but also have unique needs and concerns that reflect their online environment. Our study examines some of those similarities and differences in B2C commerce. We concentrate on new customers of a Web store and we measure their emotional and cognitive responses to determine what influences their intention to return and unplanned purchases. This is the basic nomological network that will allow us to test the metrics developed for the study.

2.1. Customer Intention to Return

Customer retention is one of the primary goals of all companies (Pine et al. 1995, Reichheld and Sasser 1990) but store loyalty for NEOs can be low because of low switching costs (anonymous 1998). Also, for those customers who view shopping as a chance for getting out, socializing, and having fun (Morris 1987), a simply functional Web site can be very unappealing. If customers cannot participate in the activities that make shopping an enjoyable experience, they may stop using Web stores and return to the more enriching and enjoyable physical world (Rice 1997, Quelch and Takeuchi 1981).

Ideally, one would measure customer loyalty by observing customers over time. Our study, however, used a cross-sectional questionnaire, so we used customer *intention* to return, not actual return visits, to measure loyalty. According to the Theory of Planned Behavior (Ajzen 1991) and its predecessor, the Theory of Reasoned Action (Fishbein and Ajzen 1975), behavioral intention can be a strong predictor of actual behavior. Therefore, we felt that intention to return is a satisfactory approximation of actual customer retention.

2.2. Why Customers Return

When shopping on the Web, consumers perform certain tasks that can elicit both emotional and cognitive responses. Those responses can determine a new consumer's intention to return for a second visit to a NEO. The double identity of the online consumer as a traditional shopper and a computer user implies that although attracting and retaining customers is a task that falls largely in the marketing realm, technology also provides tools to assist companies with that task (Straub and Watson 2001). Therefore, to understand why such a consumer returns to a store, we need to look at her interaction with the Web site both as a store and as a system.

First, we look at the Web consumer as a traditional shopper, albeit in a virtual environment. Therefore, we expect that her intention to return after a first visit will be explained by more traditional psychological/marketing variables. One model from environmental psychology states that emotional responses to the environment mediate the relationship between the environment and one's behavior (Mehrabian and Russsel 1974). According to this theory, physical and social stimuli in the environment influence the individual's emotional state. Such stimuli can be as simple as wall colors and room temperature. Also, an individual's personality traits can have an impact on his emotional state. That emotional state is defined in this model by three basic emotional responses: pleasure, dominance, and arousal. These three variables define a person's feelings that, in turn, influence behavior such as job performance and social interaction.

Another model is *flow* (Csikszentmihalyi 1975 and 1977, Csikszentmihalyi and Csikszentmihalyi 1988), defined as "the holistic sensation that people feel when they act with total involvement" (p. 36). When people are in flow, they "shift into a common mode of experience when they become absorbed in their activity. This mode is characterized by a narrowing of the focus of awareness, so that irrelevant perceptions and thoughts are filtered out, by loss of self-consciousness, by a responsiveness to clear goals and unambiguous feedback, and by a sense of control over the environment" (Csikszentmihalyi 1977, p. 72).

In the last few years, flow has also been studied in the context of information technologies and computer-mediated environments and has been recommended as a possible metric of the online consumer experience (Ghani et al. 1991, Trevino and Webster 1992, Webster et al. 1993, Ghani and Deshpande 1994, Hoffman and Novak 1996, Novak et al. 2000). While a valuable construct, we believe that flow is too broad and ill defined because of the numerous ways it has been operationalized, tested, and applied. We do, however, see value in some of the emotional and cognitive components used in flow research, namely, intrinsic enjoyment,

perceived control, and concentration/attention focus. The first two variables also correspond to those of pleasure and dominance from environmental psychology (Mehrabian and Russel 1974). These constructs can be used as valid metrics for the online consumer experience.

2.2.1. Shopping Enjoyment. A common measure of flow is the level of intrinsic enjoyment of an activity, similar to the emotional response of pleasure from environmental psychology. Enjoyment in flow has, in fact, been measured using an adapted scale from environmental psychology (Novak et al. 2000). In the context of online shopping, we operationalized intrinsic enjoyment as shopping enjoyment and measured it with a four-item scale adapted from Ghani et al. (1991).[1] Just as shopping enjoyment is important offline (Morris 1987, Forman and Sriram 1991, Blakney and Sekely 1994), it can be equally important online where it can have a significant impact on attitude and intention towards online shopping (Jarvenpaa and Todd 1997a and b, Eighmey 1997). However, while shopping in the physical world can be a very enriching and emotionally fulfilling activity, shopping on the Web does not always provide the same experience because it is limited to mostly two-dimensional pictures and text.

The importance of enjoyment in online shopping has been challenged in the past. At least one study found that recreational orientation was not different between nononline buyers, occasional online buyers, and frequent online buyers (Li et al. 1999). However, recreational orientation was defined as the consumer's general attitude towards shopping, i.e., if shopping is something they enjoy overall. In our study, we look at the effect that a *specific online* shopping experience, and its enjoyment factor, can have on consumers.

Past studies have indicated that shopping enjoyment can be an important determinant of online customer loyalty (Jarvenpaa and Todd 1997a and b, Rice 1997, Eighmey and McCord 1998). Also, flow research indicates that intrinsic enjoyment can positively impact the use of computer-mediated environments for e-mail use (Trevino and Webster 1992), other software use (Webster et al. 1993), and Web use (Novak et al. 2000). We expect the effects on using a Web store to be similar. We, therefore, hypothesize that:

HYPOTHESIS 1a. *Shopping enjoyment is positively related to intention to return.*

2.2.2. Perceived Control. Perceived control has been used in several theoretical frameworks under different variations such as perceived locus of control (Rotter 1966), perceived control in achievement motivation theory (Atkinson 1964), self-efficacy (Bandura 1982), and perceived behavioral control in the theory of planned behavior (TPB) (Ajzen 1991). In flow research, perceived control has been defined as the level of one's control over the environment and one's actions. It is similar to Bandura's self-efficacy and Ajzen's perceived behavioral control in that it is specific to an action and it can vary for different situations or actions. In the same way, self-efficacy can be task- or situation-specific and perceived behavioral control in TPB is directed towards a specific behavior. Perceived control is also similar to the emotional response of dominance from environmental psychology, where it is defined as feeling "unrestricted or free to act in a variety of ways" in a specific situation and environment (Mehrabian and Russel 1974). In fact, an adapted scale for dominance has been used to measure perceived control in flow research (Novak et al. 2000). In our study, we measured perceived control with a four-item scale adapted from Ghani et al. (1991).

For customers who shop on the Web, the information environment can be very different than for customers who shop in the physical world. The combination of less time available for shopping (Engel et al. 1990, Bellman et al. 1999), constant human cognitive resources available for information processing (Miller 1956), and an explosion of information and products available on the Web has led to more utilitarian customers demanding more control, less effort, and higher efficiency during shopping (Jarvenpaa and Todd 1997a and b, Clawson 1993, Tracy 1998). NEOs have responded to these challenges by providing site features such as search engines and recommendation agents to enable consumers to easily find what they need, learn more about it, and quickly purchase it. All

[1]A description of all the scales used in the study is available in the Appendix.

these site features result in Web customers enjoying high levels of control and convenience (Baty and Lee 1995, Hoffman and Novak 1996). Given the utilitarian nature of online consumers, we expect them to favor sites that provide them with a sense of perceived control, and so they would be more likely to remain loyal to them than other sites:

HYPOTHESIS 1b. *Perceived control is positively related to intention to return.*

2.2.3. Concentration/Attention Focus. For an individual to be in "flow," they must concentrate on their activity. Therefore, concentration has been a significant correlate or measure of flow. It can also play a role in online consumer behavior. Web customers can have a short attention span because of their limited resources of time and information processing (Engel et al. 1990, Quelch and Klein 1996, Miller 1956) together with increased levels of control (Hoffman and Novak 1997, Sheth and Sisodia 1997). However, concentration can be critical for completing their purchases efficiently.

Buying online is carried out in front of a computer at home, with numerous distractions (such as children, television, and the telephone) or at the office (where distractions include work, colleagues, and phone calls.) A consumer can also be distracted by other online activities like e-mail, instant messaging, or other Web sites. Such distractions can limit online consumer concentration. When consumers shop in the physical world, they must allocate most of their attention to that task. They must walk or drive to the stores, look through the products, interact with sales people, and make purchases. They are not able to perform any other activities such as pay their bills or answer their e-mails. On the other hand, online customers can take advantage of the multitasking capabilities of their computers and do more than shop. They can go back and forth between the Web store and their e-mail, interrupt their shopping experience to do something else in their home or office, or take advantage of download delays to perform other tasks. This can distract their attention and decrease their concentration in the purchasing task.

Concentration as a measure of flow has been found to positively influence the overall experience of computer users (Novak et al. 1998) and their intention to use a system repeatedly (Webster et al. 1993). Also, we know that interruptions that limit concentration reduce Web users' satisfaction with online shopping (Xia and Sudharshan 2000). We therefore expect that high concentration, measured with a four-item scale adapted from Ghani et al. (1991), would have a positive impact on intention to return:

HYPOTHESIS 1c. *Concentration is positively related to intention to return.*

However, the online consumer is also a computer user. Our model would be incomplete if we did not account for the reasons computer users adopt a technology such as a specific Web site. So, we turn to the most widely used theoretical model for explaining system usage, the Technology Acceptance Model (TAM) (Davis 1989). TAM has been tested in many empirical studies that include user acceptance of word processors (Davis et al. 1989), spreadsheets (Mathieson 1991), e-mail (Szajna 1996), voice mail (Straub et al. 1995), and telemedicine technology (Hu et al. 1999). By treating a Web store as a technology system and the Web consumer as a computer user, we can apply TAM and test how well it predicts user intention to use the technology, i.e., the Web store. In our case, that would translate into customer intention to return to the store. According to TAM, attitude towards technology affects use of the technology. Two belief variables have an impact on attitude: perceived ease of use and perceived usefulness of the technology. Empirical research has shown that attitude towards the technology is not a significant mediating variable (Venkatesh and Davis 1996, Venkatesh 1999) so we do not include it.

We believe that the TAM variables can also be successfully applied in the context of online consumer behavior. Viewing the store as a Web site or system, we expect that when customers believe that using the Web site will enhance their shopping productivity (perceived usefulness) they will be more likely to return. We anticipate the same effect when customers believe that the Web site is easy to use (perceived ease of use):

HYPOTHESIS 2a. *Perceived usefulness of the Web store is positively related to intention to return.*

HYPOTHESIS 2b. *Perceived ease of use of the Web store is positively related to intention to return.*

While TAM, environmental psychology, and flow theory may seem too discrete to combine in one study, we believe that they can complement each other successfully to form a hybrid theoretical perspective (Shaw and Jarvenpaa 1997). First, environmental psychology and flow share the common variables of enjoyment/pleasure and perceived control/dominance, the emotional responses of this study. The TAM variables (perceived usefulness and perceived ease of use) and concentration provide the cognitive responses. All three theories have at their core a number of emotional and cognitive responses to the environment that influence an individual's behavior. The context of online shopping and the dual nature of the Web customer as a computer user allows them to come together to help us better understand online consumer behavior.

2.3. Unplanned Purchases

While customers may enter a store planning to make specific purchases, they often end up making unplanned purchases. There are four types of unplanned purchases (Stern 1962):

(1) Pure impulse: These are purchases that are made for purely hedonic reasons and are usually characterized by:

(a) spontaneity;

(b) power, compulsion, and intensity;

(c) excitement and stimulation;

(d) disregard for consequences (Rook 1987).

(2) Reminder effect: A stimulus reminds the consumer to buy a product he needs.

(3) Suggestion effect: The customer purchases a product because of a promotion.

(4) Planned impulse: The customer shops without any specific product(s) in mind.

Unplanned purchasing behavior on the Web can be interesting and complex. While shopping trips in the physical world are often constrained by time and geographical location, Web consumers can buy at any time from anywhere, possibly increasing the number of products they buy on impulse. The ability to click on banner ads and be transported immediately to the store, in front of the product advertised could make it easier for consumers to make unplanned purchases. Finally, the Web provides more privacy than the physical world, thereby possibly enabling consumers to indulge in impulsive shopping behavior they would find embarrassing offline.

Some of the same emotional and cognitive responses that impact a customer's intention to return should influence the number of unplanned purchases made. We do not believe, however, that the belief variables of TAM will determine unplanned purchases. TAM predicts intention to use a technology and subsequently the amount of usage. It does not differentiate between specific kinds of use or the results of such use, e.g., making unplanned purchases at a Web store.

We know from flow research that intrinsic enjoyment can increase a user's exploratory behavior (Ghani and Deshpande 1994). Also, in their comprehensive study of impulse buying, Beatty and Ferrel (1998) found that in-store browsing behavior might increase the urge to buy impulsively. If online consumers enjoy their shopping experience, they might engage in more exploratory browsing in the Web store leading to more unplanned purchases. In addition, Beatty and Ferrel found an increase in impulse purchasing urges for shoppers with positive feelings during shopping. Beatty and Ferrel's work has shown the significance of positive emotional responses for unplanned purchases. Their impulsive nature implies that they rely a lot on consumer feelings. So, we believe that shopping enjoyment will induce unplanned purchases online.

HYPOTHESIS 3a. *Consumers with higher shopping enjoyment are more likely to make unplanned purchases.*

Strategically placed promotions and displays throughout a store, can increase unplanned purchases and overall sales (Inman et al. 1990). Online, customers have high levels of control over what they see and do through the use of search engines, intelligent agents, and recommender systems (Baty and Lee 1995, Hoffman and Novak 1996 and 1997, Sheth and Sisodia 1997). This increased control allows consumers more choice over the advertising and promotional material they are exposed to, reducing unplanned purchases (Draft 1993, Rust and Oliver 1994, Shell 1994, Raman and Leckenby 1995, Burke 1997, Raman 1997, Lohse and Spiller 1998a).

In this study, we measure *perceived* control. If it is

equivalent to *actual* consumer control, then, as discussed above, it may lead to a decrease in unplanned purchases. Alternatively, the *perception* of being in control could contribute to a positive experience that could increase unplanned purchases. We believe that *perceived* control online would reflect *actual* control, so we expect that it would impact unplanned purchases negatively.

HYPOTHESIS 3b. *Consumers with higher perceived control are less likely to make unplanned purchases.*

We know that exposure to marketing promotions increases unplanned purchases (Assael 1992). Consumers that are able to focus their attention at a Web store should also be more likely to notice marketing promotions on the site. If consumers are not paying full attention to the contents of the Web site when buying online, they are less likely to notice products that they might otherwise buy on impulse.

HYPOTHESIS 3c. *Consumers with higher concentration are more likely to make unplanned purchases.*

2.4. Determinants of Emotional and Cognitive Responses

Once we know how certain emotional and cognitive responses to a Web store effect a consumer's purchasing behavior, the next logical step is figuring out how to elicit the right responses. To gain insight, the study looked at the effect of several variables that can determine a consumer's emotional and cognitive responses.

2.4.1. Product Involvement. Product involvement and its measurement have been the source of considerable research and debate since Personal Involvement Inventory (PII) was first proposed and analyzed by Zaichkowsky (1985). While there have been many variations on the definition of involvement (Zaichkowsky 1985, Greenwald and Leavitt 1984, Mitchell 1981, Park and Mittal 1985), it is generally accepted that involvement is: (a) a person's motivational state (i.e., arousal, interest, drive) towards an object where (b) that motivational state is activated by the relevance or importance of the object in question (Mittal 1989). In the current study, the Revised Personal Involvement Inventory (RPII) proposed by McQuarrie and Munson (1992) was used to measure product involvement for books, i.e., how interested consumers are in books and how important books are to them.

While involvement with advertisements (Andrews and Durvasula 1991) and with the purchase process (Slama and Tashchian 1985) is important, the study focuses on involvement with the product. We are not concerned with how consumers are attracted to a Web site but with what happens while they visit it. Also, our three emotional and cognitive responses of perceived control, shopping enjoyment, and concentration give us a good indication of the involvement of the consumer with the purchase process. As a result, we only measure product involvement.

A consumer's involvement with the product(s) sold by a Web-based company can have an effect on the consumer's experience and behavior. Involvement, measured simply as importance of the Web to the consumer, also had a strong effect on the primary antecedents of flow (Novak et al. 2000). We expect that customers with higher product involvement will have a more positive shopping experience due to their increased interest in the product. This relationship should be true for shopping enjoyment and concentration but not for perceived control. Perceived control is more of a result of the interaction with the Web site and its features, rather than the type of product purchased.

HYPOTHESIS 4a. *Product involvement is positively related to shopping enjoyment.*

HYPOTHESIS 4b. *Product involvement is positively related to concentration.*

2.4.2. Demographics. Demographic variables have been used extensively in market research to segment the customer population for better marketing strategies (Engel et al. 1990, Assael 1992). In some cases, demographics variables may determine some consumer behavior on the Web (Korgaonkar and Wolin 1999). For parsimony, we do not hypothesize on any possible differences in online consumer emotional and cognitive responses due to gender or age.

2.4.3. Web Skills. One of the most important antecedents of flow is the level of skills of the individual (Csikszentmihalyi 1975 and 1977, Ghani et al. 1991, Trevino and Webster 1992, Webster et al. 1993, Ghani

and Deshpande 1994, Hoffman and Novak 1996, Novak et al. 2000). Skills are measured as perceived by the user and not through observation or a standardized test. Therefore, they are similar to computer self-efficacy, defined as "an individual judgment of one's capability to use a computer" (Compeau and Higgins 1995). Computer self-efficacy has been found to affect computer use, often through its effect on the emotional state of the user by, for example, reducing his computer anxiety (Marakas et al. 1998). Similarly, skills have been consistently found (along with challenges, discussed below) to be significant antecedents to flow. Therefore, we expect, that as online consumers perceive their Web skills to be higher, they will be more likely to have positive emotional and cognitive responses to the Web store they visit.

HYPOTHESIS 5a. *Perceived skills are positively related to perceived control.*

HYPOTHESIS 5b. *Perceived skills are positively related to shopping enjoyment.*

HYPOTHESIS 5c. *Perceived skills are positively related to concentration.*

2.4.4. Product Search Mechanisms. Web stores have responded to the call for customer control by providing various site features like internal search engines and recommender systems to enable consumers to easily find what they need, learn more about it, and quickly purchase it. There have been very few attempts to create a framework or typology of Web-based customer decision-support systems (O'Keefe and McEachern 1998). We differentiated search mechanisms according to the two types of information they use: nonvalue-added and value-added.

Nonvalue-added information consists of all information that is publicly available and standardized, such as objective information that describes the product sold. For a bookstore, such information can be the book title, author, and publisher. *Value-added information* consists of all information that is generated by the NEO. It is subjective and often proprietary to the entity that generates it. Value-added information can be created by the NEO (such as a weekly bestseller list), from a third party (such as hyperlinks to online reviews of products by independent parties or Oprah's Book Club selections), or from the customers themselves (such as customer reviews available to other customers).

We know that product search may be undertaken simply because it is fun as in window shopping. Also, customers often engage in ongoing information gathering independent of specific needs or purchase decisions (Bloch et al. 1986). For them, value-added information can prove interesting and helpful. Whether it is reviews of books by other customers or stock recommendations by analysts, customers can enjoy this wealth of information about products and services not always available in the physical world. In general, we expect that the more enriched and satisfying the shopping experience is, the more likely customers are to experience high levels of perceived control, shopping enjoyment, and concentration. Because product information search can be a fun-seeking experience (Bloch et al. 1986), we expect that the use of value-added search mechanisms can make the shopping experience more fulfilling and enjoyable. Access to such information can improve consumer decision making through complex, nonlinear, and nondirected queries (Hoffman et al. 1995) and can be an important incentive for people to shop online (Jarvenpaa and Todd 1997a and b) by providing more control over their purchases. Such mechanisms may also help customers concentrate better because they require active customer participation and interpretation of the value-added information.

HYPOTHESIS 6a. *The use of value-added search mechanisms is positively related to perceived control.*

HYPOTHESIS 6b. *The use of value-added search mechanisms is positively related to shopping enjoyment.*

HYPOTHESIS 6c. *The use of value-added search mechanisms will be positively related to concentration.*

2.4.5. Challenges. Along with individual skills, the challenges presented by an activity are the most important predictors of flow (Csikszentmihalyi 1975 and 1977, Ghani et al. 1991, Trevino and Webster 1992, Webster et al. 1993, Ghani and Deshpande 1994, Hoffman and Novak 1996, Novak et al. 2000). Positive challenges from the shopping experience on the Web can affect consumer responses to that experience. It is

important to stress here that we are looking at *positive* challenges, such as those presented by a sport or a game. We are not looking at negative challenges that are due to problems like difficult navigation, slow download time, or product unavailability. A Web site can be perceived by a consumer as positively challenging in the same way an electronic game is challenging. The process of discovering information and using special Web site features can feel challenging and ambitious with positive effects.

HYPOTHESIS 7a. *The level of challenges of a Web store is positively related to perceived control.*

HYPOTHESIS 7b. *The level of challenges of a Web store is positively related to shopping enjoyment.*

HYPOTHESIS 7c. *The level of challenges of a Web store is positively related to concentration.*

The complete theoretical framework (Figure 1) is also a nomological network used to test the construct validity of our variables. Constructs are considered nomologically valid if, as measured in our study, they predict or are predicted by other constructs consistent with past research (Cronbach 1971, Bagozzi 1990, Straub et al. 1995).

3. Methods

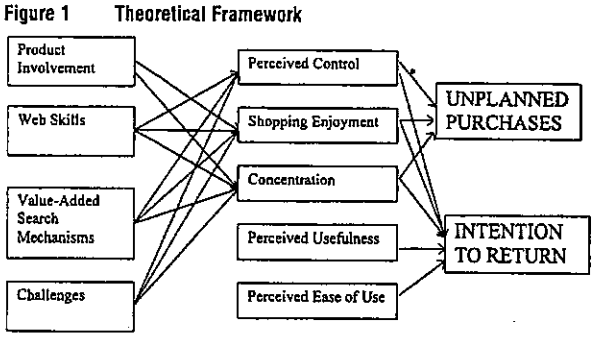
To test our nomological network, an empirical study using an online questionnaire was administered to actual Web customers. The Web store selected for this study is Booksamillion.com, the online branch of the traditional "brick and mortar" company Books-A-Million. Booksamillion.com was selected for several reasons:

(1) The site is comprehensive and functional, eliminating any Web site problems.

(2) Booksamillion.com is a relatively unknown online bookseller that guaranteed us enough first-time customers for our sample.

(3) The site, unlike booksellers like Amazon.com, primarily sells books. Books are a commodity—a fact that helps eliminate variance in product quality (the actual physical book, not the quality of its content). Though the site sells a few other items, such as coffee and magazines, the main focus of the store is books.

Figure 1 Theoretical Framework



* All relationships are hypothesized positive except Perceived Control → Unplanned Purchases, which is negative.

3.1. Study Design

Obtaining permission to directly survey Booksamillion.com customers proved to be too difficult. Companies are reluctant to participate in research projects that interrupt customers' shopping or the firm's internal operations. So, we simulated actual customers by recruiting subjects, asking them to visit the Booksamillion.com site and then having them answer our questionnaire. The fact that subjects did not visit the Web site on their own volition is definitely a limitation of the study. However, while it is possible that their intention to return would differ when directed to visit the site by the researcher, we do not believe that the difference is large enough to compromise the generalizability of our results.

To measure unplanned purchases we needed to know which purchases consumers had initially planned to make when they visited the store. We simulated an initial purchase plan by providing subjects with a $10 gift certificate. Subjects were asked to fill out the first part of the questionnaire for a $10 gift certificate to an online bookstore, which they would visit upon completion. The first part asked them questions unrelated to their particular visit, including how many (if any) books they intended to buy when they visited the online bookstore. The $10 gift certificate ensured us that some of the subjects would form an initial purchase plan of spending the gift certificate at the store.

After visiting the site, we asked the subjects to immediately return and answer the second part of the questionnaire, where we asked about their shopping

experience and their actual purchases. Any purchases made in addition to what they had initially said they would buy were unplanned purchases. Unplanned purchases were calculated using the *number* of books purchased over their initial purchase plan, instead of the *amount* (in dollars) spent over the $10 gift certificate. We felt that it was a more accurate measure because customers most often go to a store with a plan to buy a number of books and not to spend an amount of money. Because our hypotheses deal with the *likelihood* of making unplanned purchases, we created a binary variable with the value zero if a customer made no unplanned purchases and the value one if a customer made some unplanned purchases.

3.2. Implementation

The sample was recruited from the database of Dynamic Logic, Inc., an online market research firm that specializes in the measurement of the effectiveness of online ad. campaigns through the use of online questionnaires. Dynamic Logic also hosted the questionnaire using its patented technology that features randomized order of questionnaire items (to eliminate fatigue effect), live online results, and automated data collection.

The sample was randomly selected from the Dynamic Logic database of registered participants who were e-mailed an invitation to participate in an online questionnaire for a $10 gift certificate to an online bookstore, $10 in cash (via check), and the chance to win $1,000 in cash. Each subject was asked which online bookstores he had visited in the past and if Books amillion.com was among them; the subject was screened out, leaving only new customers. They filled out the first part of the questionnaire and were presented with a gift certificate code and a button. When they clicked it, a new window opened with the Books amillion.com site. They were instructed to visit Books amillion.com and then immediately return to the questionnaire to fill out the second part for an additional $10 in cash and a chance to win $1,000. We used a cash incentive for the second part of the questionnaire, instead of another gift certificate, so as not to influence the respondents' answer to the question on their intention to return to the Web store.

Our research instrument asked about the consumers' experience during that specific visit to the store. This increased the validity of our measures. Because we were not working with Booksamillion.com, we were unable to use log files to measure behavioral variables such as the number of purchases. We relied on self-reports for such measures. Subjects were not allowed to proceed unless they filled out the entire questionnaire (see Appendix C).

3.3. Sample and Descriptive Statistics

The questionnaire ran for one week. The first part of the questionnaire was completed by 300 subjects, and 280 of those filled out the second part, a retention rate of 93.3%. All 280 subjects were new customers to Books amillion.com. There were 12 international respondents, mainly from Canada. 63% of the sample was female, and about 50% of the subjects were 30 to 50 years old. 77% of the subjects had at least some college education, and 48% reported an income of over $50,000. Almost half of the subjects reported having children under the age of 18, and 68% reported spending over 15 hours per week online. Almost 50% of the subjects reported buying books online at least four times a year. When asked to rate loyalty to a specific online bookstore on a scale from one to six, many subjects (over 45%) reported low customer loyalty (values of one or two on a six-point scale).

The majority of respondents (73.2%) reported that they intended to buy no books or that they did not know how many they would buy. There were 20% who wanted to purchase one book. After visiting the site, almost half of the subjects said they purchased at least one book, with almost 10% purchasing more than one. About 32% of the subjects made an unplanned purchase; 24% bought one more book than planned; 6% bought two more; and four subjects bought more than two extra books. Very few purchased cross-selling items such as coffee (4 subjects) or magazines (17 subjects). Many respondents (65%) spent 20 minutes or less at Booksamillion.com, indicating an efficient purchasing process. Subjects seemed relatively satisfied with their experience, with 33% reporting they would return to Booksamillion.com (giving six or seven on a seven-point scale). Only 14% said that they would not return (giving one or two on the same scale).

3.4. Results

Based on prior research, we expected the flow factors as well as the TAM factors to be correlated. Therefore,

we factor analyzed our scales using principal components extraction with direct oblimin rotation, an appropriate method when there is reason to expect the factors to be correlated (Pedhazur and Pedhazur 1991). This analysis validated our instrument and the multi-item scales, as shown in the Appendix. All scales had acceptable Cronbach's alpha values (Nunnally 1967), as seen in Table 1.

The testing of the hypotheses that follows is intended to demonstrate further validation of the instrumentation, as discussed earlier. If the constructs perform as predicted by theory, then we can infer that the measurement of the constructs is nomologically valid.

In testing the model, we were interested in seeing the difference in explanatory power between the part of the model that considers the online consumer as a shopper and that which considers the consumer as a computer user. Therefore, we tested the three flow variables separately from the TAM variables. We then tested the TAM variables independently. Finally we tested all variables together in a single model.

Table 2 shows the linear regression model with the three flow variables. After removing four outliers that had standardized residuals with absolute values greater than 3.0 (Chatterjee et al. 1995), the final model had an excellent fit and explained a large percentage of variance with a $R^2 = 0.472$. Only the coefficient for shopping enjoyment was statistically significant. Therefore, Hypothesis 1a is supported but Hypotheses 1b and 1c are not.[2]

Table 3 shows the linear regression model with the TAM variables. After removing the same four outliers, the model had a great fit and explained about the same variance ($R^2 = 0.489$). Only the coefficient for perceived usefulness was significant, while that for perceived ease of use was not. Therefore, only Hypothesis 2a is supported.

Comparing the two models, we saw very little difference in their predictive ability for intention to return. We ran a combined regression model, as seen in Table 4, after removing the same four outliers. The model had good fit and, as expected, explained more variance ($R^2 = 0.546$). The same coefficients for shopping enjoyment and perceived usefulness were significant, so the same hypotheses are supported.

We ran a logistic regression on the binary variable of unplanned purchases to test the effects of perceived control, shopping enjoyment, and concentration. Table 5 shows the results. The chi-square value shows that

Table 1 **Summary Statistics and Cronbach's Alpha Values for All Scales**

	Mean	S.D.	Cronbach's Alpha
Concentration	4.52	1.46	0.910
Shopping Enjoyment	4.30	1.52	0.944
Perceived Control	4.98	1.52	0.813
Skills	5.31	1.50	0.918
Challenges	2.96	1.44	0.803
Involvement	6.13	1.01	0.929
Perceived Ease of Use	5.16	1.61	0.927
Perceived Usefulness	4.15	1.57	0.924
Intention to Return	4.59	1.77	N/A
Value-Added Used	1.37	1.31	N/A

[2]For all regression models in the study, low VIF values indicate low collinearity, the standardized residuals are normally distributed, and there is no heteroscedasticity.

Table 2 **Linear Regression Model for Intention to Return with the Flow Variables**

Variable	B	Std. Error	β
Constant	0.866**	0.327	
Perceived Control	0.104	0.064	0.090
Shopping Enjoyment	0.705**	0.080	0.608**
Concentration	0.045	0.070	0.037

$N = 276$; $R^2 = 0.472$; $F = 81.069$ ($p < 0.01$)—where ** $p < 0.01$, * $p < 0.05$, ° $p < 0.1$

Table 3 **Linear Regression Model for Intention to Return with the TAM Variables**

Variable	B	Std. Error	β
Constant	1.130**	0.259	
Perceived Usefulness	0.695**	0.066	0.621**
Perceived Ease of Use	0.119°	0.064	0.109°

$N = 276$; $R^2 = 0.489$; $F = 130.651$ ($p < 0.01$)—where ** $p < 0.01$, * $p < 0.05$, ° $p < 0.1$

Table 4 Linear Regression Model for Intention to Return with All Variables

Variable	B	Std. Error	β
Constant	0.701*	0.308	
Perceived Control	0.021	0.069	0.018
Shopping Enjoyment	0.400**	0.088	0.345**
Concentration	0.040	0.066	0.033
Perceived Usefulness	0.465**	0.074	0.415**
Perceived Ease of Use	−0.002	0.073	−0.002

$N = 276$; $R^2 = 0.546$; $F = 64.986$ ($p < 0.01$)—where $p < 0.01$, * $p < 0.05$, ° $p < 0.1$

the model has good fit but none of the coefficients is significant. So, Hypotheses 3a, b, and c are not supported.

We used linear regression models to test the effect of individual and environmental factors on perceived control, shopping enjoyment, and concentration. Table 6 shows the results of the regression model for concentration. The model had good fit and explained about 21% of the variance. The coefficients for product involvement, challenges, and skills were significant and positive, providing support for Hypotheses 4b, 7c, and 5c. The coefficient for value-added use was in the right direction, but not significant.

The regression model for shopping enjoyment had an even better fit and explained more variance with a $R^2 = 0.281$, as seen in Table 7. The coefficients for product involvement, challenges, skills, and value-added use were highly significant and positive, providing support for Hypotheses 4a, 7b, 5b, and 6b, respectively.

The regression model for perceived control had a moderately good fit ($F = 2.898$, $p = 0.006$) but no significant coefficients and a $R^2 = 0.070$ that indicated the model explained too little variance. Therefore, we could not verify Hypotheses 7a and 8a.

Figure 2 shows our nomological network but includes only the relationships that were supported by our data.

4. Discussion

An important contribution of our study is the testing and validation of metrics for consumer behavior on the

Table 5 Logistic Regression for Unplanned Purchases with the Flow Variables

	B	Std. Error	Exp (B)
Constant	−2.974	0.614	0.051
Concentration	0.206°	0.123	1.229°
Shopping Enjoyment	0.142	0.139	1.152
Perceived Control	0.123	0.111	1.130

Initial −2 Log Likelihood = 350.136
−2 Log Likelihood = 332.413
Chi-square = 17.724**
df = 3
$N = 280$; Cox and Snell $R^2 = 0.061$; Nagelkerke $R^2 = 0.086$—where ** $p < 0.01$, * $p < 0.05$, ° $p < 0.1$

Table 6 Linear Regression Model for Customer Concentration

Variable	B	Std. Error	β
Constant	0.047	0.646	
Involvement	0.407**	0.083	0.280**
Challenges	0.221**	0.056	0.216**
Skills	0.138*	0.056	0.142*
Value-Added Use	0.130°	0.070	0.116°
Nonvalue-Added Use	0.088	0.074	0.074
Gender	0.014	0.176	0.005
Age	0.008	0.007	0.066

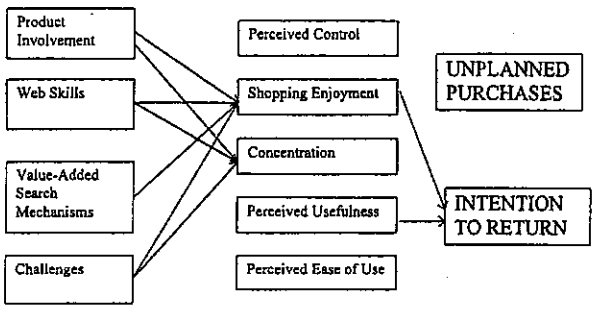
$N = 280$; $R^2 = 0.210$; $F = 10.321$ ($p < 0.01$)—where ** $p < 0.01$, * $p < 0.05$, ° $p < 0.1$

Table 7 Linear Regression Model for Customer Shopping Enjoyment

Variable	B	Std. Error	β
Constant	0.086	0.640	
Involvement	0.328**	0.082	0.218**
Challenges	0.358**	0.055	0.338**
Skills	0.182**	0.055	0.180**
Value-Added Use	0.240**	0.069	0.207**
Nonvalue-Added Use	0.042	0.073	0.034
Gender	0.171	0.174	0.054
Age	−0.011°	0.007	−0.090°

$N = 280$; $R^2 = 0.281$; $F = 15.095$ ($p < 0.01$)—where ** $p < 0.01$, * $p < 0.05$, ° $p < 0.1$

Figure 2 Relationships Supported by Empirical Study



All relationships are statistically significant at the $p < 0.05$ level.

Web as called for by Straub and Watson (2001). In addition to the fact that all the scales used in the study showed high reliability, those of shopping enjoyment, perceived usefulness, involvement, challenges, and skills also demonstrated high nomological validity. When tested in a nomological network derived from prior research, they generally behaved as demonstrated in past studies. Therefore, we believe that future research in online consumer behavior can use these metrics with some assurance. Testing them against other important factors not included in this study will advance our understanding of online consumer behavior.

From a substantive point of view, we examined how emotional and cognitive responses to the shopping experience on the Web can influence online consumer behavior, specifically, intention to return and unplanned purchases for new customers. Looking at the online consumer both as a shopper and a computer user, by blending theoretical paradigms from psychology, marketing, and information systems, we confirmed the dual nature of the online consumer as a traditional shopper and a computer user. Our results show that both enjoyment of the shopping experience (a psychological/marketing variable) and perceived usefulness of the Web site (an IS variable for system use) are important for a new customer's intention to return.

Furthermore, the significance of perceived usefulness shows that the Technology Acceptance Model can be successfully applied, even when the behavior in question is not one of pure system usage. When tested alone, without the flow variables, perceived usefulness explained 49% of variance of intention to return, demonstrating even further the robustness of TAM in this context. Our results also confirmed prior TAM research that found that perceived usefulness was a more important predictor of intended system usage than perceived ease of use (Davis 1989, Hu et al. 1999).

In accordance with research that has demonstrated the utilitarian nature of online consumers (Jarvenpaa and Todd 1997a and b, Clawson 1993), we found that a consumer's belief about the usefulness of a Web store (a cognitive response) can determine their future visits. However, we also found that an emotional response to the Web site can also have the same effect. Even though consumers may not *expect* to be entertained when they shop online, if they do enjoy their experience, they are more likely to return to the Web store. These results show that online consumers are not purely utilitarian, valuing only efficiency in shopping, but they can also enjoy shopping online enough to make them return. If this means that online consumers are more similar to offline ones than previously thought, researchers of online consumers may successfully use previous research done offline, though with caution.

Our results on unplanned purchases were surprising. We found no relationship between unplanned purchases and the flow variables of shopping enjoyment, concentration, and perceived control. A possible reason for the weak results could be our $10 gift certificate. One could argue that it is difficult to purchase anything with $10, even if it is a book. This may have prevented some of our subjects from buying anything at all in the first place. Unfortunately, the high cost of such an incentive made it infeasible for our study. It is also possible that there are other variables that can explain unplanned purchases that we did not include in our framework. Future research needs to reexamine the relationships we tested as well as introduce new possible predictors.

Our results also confirmed some prior research on flow. We found that perceived Web skills and positive challenges are positively related with shopping enjoyment and concentration of online consumers. The importance of Web skills is also consistent with work done in computer self-efficacy (Marakas et al. 1998)

and stresses once again the "computer user" part of the dual identity of the online consumer. The more confident and comfortable consumers feel with the Web site, the more likely it is that they will enjoy it.

The nonsignificance of the relationship of flow variables with unplanned purchases and intention to return (with the exception of shopping enjoyment) puts into question the use of flow in the context of online shopping. Unlike the study by Novak et al. (2000), which defined and measured the flow state for general Web users, we examined the effect of flow variables on consumer behavior during a *particular* store visit. It may be the case that for online consumers, not simply Web users, a multidimensional flow construct does not explain their behavior, while a simple construct like shopping enjoyment does. Therefore, we urge cautious use of flow in online consumer behavior research.

We also found that the use of value-added search mechanisms may influence the experience of online consumers that can in turn affect their behavior. The use of a utilitarian tool such as a search engine is positively related with shopping enjoyment, an emotional response. However, we found no relationship between using value-added search mechanisms and perceived control or concentration. A possible implication is that Web site features that are intended to be mere tools for the consumer to use might have a strong emotional impact. An alternative explanation is that the more consumers enjoy themselves, the more they use value-added search mechanisms.

Our results also have important practical implications for companies. The dual nature of the online consumer as a traditional shopper and as a computer user means that appropriate interface, navigational structure, and other elements of human-computer interaction may be just as important to retaining customers as good customer service and lower prices. Many Web-based companies have taken large steps to increase the convenience to their customers by providing features like express checkouts and recommender systems. However, our results have shown that emotional experiences such as shopping enjoyment can help retain customers. Therefore, online stores should provide both utilitarian value as well as hedonic value to their customers, partly by providing them with value-added search mechanisms.

Our study was not without limitations. Because our subjects had registered with an online market research company and took multiple online questionnaires, they may be more Web-savvy than the average user. Unfortunately, to recruit subjects online, they must be comfortable enough with the Web to be recruited and to participate. Our "Web skills" scale shows a high mean of 5.3 on a seven-point scale, but also a high variance of 2.25. That justifies some cautious generalizations of our results. The same applies to our scale for "perceived ease of use" with a mean of 5.16 and a variance of 2.6.

An important variable missing from our framework is planned purchases of first-time customers. How do customers decide to visit a specific Web store for a planned purchase, and what factors determine whether they will actually make that purchase at that store? Our study design did not capture actual customers as they came to a store for the first time. Instead, we had to simulate new customers by directing our subjects to the Web store. A future study that captures actual new customers would be able to better explain the factors behind customer acquisition and planned purchasing.

There are also many additional individual and environmental factors that can determine a consumer's emotional and cognitive responses. In environmental psychology, physical stimuli like colors as well as personality traits can influence an individual's emotional state (Mehrabian and Russsel 1974). In marketing, there have been studies on a variety of individual characteristics such as motivation and knowledge as well as a broad range of environmental variables including family, culture, and social class (Engel et al. 1990). There has also been some research on the antecedents of the two belief variables in the TAM (e.g., Venkatesh and Davis 1996). Our study did not consider any of these variables and we urge other researchers to do so.

We consider this study a stepping stone on the road to understanding online consumer behavior. Many issues remain unresolved and many questions unanswered. We took a first step at studying the consumer experience on the Web and found a number of metrics to be reliable and nomologically valid. Future research can use these metrics to explain how and why consumers think, act, and feel when shopping on the Web.

5. Appendices

Appendix A. Structure Matrix from Principal Component Analysis with Direct Oblimin Rotation

	FACTORS						
ITEMS	1 (Concentration)	2 (Enjoyment)	3 (Perceived Control)	4 (Challenges)	5 (Perceived Usefulness)	6 (Perceived Ease of Use)	7 (Skills)
... I was absorbed intensely in the activity.	−0.886						
... My attention was focused on the activity.	−0.861						
... I concentrated fully on the activity.	−0.860						
... I was deeply engrossed in the activity.	−0.883						
... I found my visit interesting.		0.559					
... I found my visit enjoyable.		0.662					
... I found my visit exciting.		0.648					
... I found my visit fun.		0.658					
... I felt confused.			0.473				
... I felt calm.			0.929				
... I felt in control.			0.862				
... I felt frustrated.			0.635				
... challenged me to perform to the best of my ability				0.872			
... provided a good test of my skills				0.842			
... stretched my capabilities to the limits				0.806			
Using the site can improve my shopping performance.					0.928		
Using the site can increase my shopping productivity.					0.914		
Using the site can increase my shopping effectiveness.					0.932		
I find using the site useful.					0.725		
Learning to use the site would be easy for me.						0.899	
My interaction with the site is clear and understandable.						0.814	
It would be easy for me to become skillful at using the site.						0.898	
I find the site easy to use.						0.899	
I am very skilled at using the Web.							0.945
I know how to find what I want on the Web.							0.914
I know more about using the Web than most users.							0.914

Appendix B. Correlation Matrix

Pearson Bivariate Correlations

	Perceived Ease of Use	Perceived Usefulness	Perceived Control	Enjoyment	Concentration	Skills	Involvement	Value-Added Used	Intention to Return
Challenges	0.01	0.39**	−0.01	0.34**	0.23**	−0.10	0.02	0.06	0.27**
Perceived Ease of Use		0.68**	0.70**	0.66**	0.43**	0.24**	0.31**	0.22**	0.47**
Perceived Usefulness			0.55**	0.75**	0.47**	0.14*	0.19**	0.21**	0.62**
Perceived Control				0.58**	0.25**	0.06	0.12*	0.15*	0.39**
Enjoyment					0.65**	0.18**	0.27**	0.28**	0.62**
Concentration						0.14*	0.33**	0.21**	0.39**
Skills							0.09	0.06	0.09
Involvement								0.12*	0.18**
Value Added									0.19**

where ** $p < 0.01$, * $p < 0.05$

Appendix C. Survey Instrument

1. Intention to Return

We measured customer intention to return with the following single item on a seven-point Lickert scale ranging from one—extremely unlikely to seven—extremely likely:

"How likely is it that you will visit Booksamillion.com again in the future?"

2. Unplanned Purchases

We used two questions to determine the number of unplanned purchases. The first question asked how many books the customer intended to buy when she would be directed to the bookstore site with the $10 gift certificate. The possible answers were none, one, two, three, four or more, and "I don't know how many" books. The second question asked how many books they actually bought as well as how many of the other items sold on Booksamillion.com, such as coffee and magazines, they purchased.

We calculated the number of unplanned purchases by subtracting the first question from the second one. For those subjects that reported "I don't know how many books" in the first question, we considered their initial planned purchases as zero. We then created a binary variable that takes the value zero for subjects who did not make unplanned purchases and the value one for those who did.

3. Search Mechanisms Used

A multiple response question was used to determine which search mechanisms the customer used, and a single response question asked customers which one of the search mechanisms was the most helpful. Once again, due to the lack of resources, we depended on self-reports rather than Web logs to determine which search features customers used. We separated the search mechanisms available to the customers into value-added and nonvalue-added. We then created a variable that measured the use of each type of search mechanism by adding the number of search mechanisms the customers reported they had used.

Value-Added Search Mechanisms	Nonvalue-Added Search Mechanisms
Bestsellers	Book Search
Bargains	Browse Subjects
Reviews	Upcoming Titles
Weekly Specials	All Authors
New and Notable	Road to El Dorado
Easter	Left Behind Series
Great Movie Reads	eBook: Riding the Bullet
Campaign 2000	Star Wars
Oprah's Book Club	Pokemon
Seen in NY Times	MLB Showdown 2000
Graduation	Bible Covers
Mother's Day	Accessories
Harry Potter Fans	Magic the Gathering
Autographed Books	Yo Yos
Popular Business	Furby
All Literary Awards	Tae Bo
Music Bestsellers	Coin Collectibles
Tender Times	Don't Sweat Series
	Coffee
	Magazines

Classification of search mechanisms on Booksamillion.com

4. Multi-Item Scales

Following are the multi-item scales used in the study and the source in prior literature from which they were adapted. All items used a seven-point Lickert scale:

Concentration/Attention Focus (Ghani et al. 1991):

During my last visit to Booksamillion.com . . .

. . . I was absorbed intensely in the activity.

. . . My attention was focused on the activity.

. . . I concentrated fully on the activity.

. . . I was deeply engrossed in the activity.

Shopping Enjoyment (Ghani et al. 1991):

During my last visit to Booksamillion.com . . .

. . . I found my visit interesting.

. . . I found my visit enjoyable.

. . . I found my visit exciting.

. . . I found my visit fun.

Perceived Control (Ghani et al. 1991):

During my last visit to Booksamillion.com . . .

. . . I felt confused *(reversed).*

. . . I felt calm.

. . . I felt in control.

. . . I felt frustrated *(reversed).*

Web Skills (Novak et al. 1998):

I am very skilled at using the Web.

I know how to find what I want on the Web.

I know more about using the Web than most users.

Challenges (Novak et al. 1998):

Using Booksamillion.com challenged me to perform to the best of my ability.

Using Booksamillion.com provided a good test of my skills.

Using Booksamillion.com stretched my capabilities to the limits.

Perceived Usefulness (Venkatesh and Davis 1996):

Using Booksamillion.com can improve my shopping performance.

Using Booksamillion.com can increase my shopping productivity.

Using Booksamillion.com can increase my shopping effectiveness.

I find using Booksamillion.com useful.

Perceived Ease of Use (Venkatesh and Davis 1996):

Learning to use Booksamillion.com would be easy for me.

My interaction with Booksamillion.com is clear and understandable.

It would be easy for me to become skillful at using Booksamillion.com.

I find Booksamillion.com easy to use.

Product Involvement (McQuarrie and Munson 1992):

We would like to know how interested you are in books. Please use the series of descriptive words listed below to indicate your level of interest in books:

Important	O O O O O O O	Unimportant
Irrelevant	O O O O O O O	Relevant
Means a lot to me	O O O O O O O	Means nothing to me

Unexciting	O	O	O	O	O	O	O	Exciting
Dull	O	O	O	O	O	O	O	Neat
Matters to me	O	O	O	O	O	O	O	Doesn't matter to me
Boring	O	O	O	O	O	O	O	Interesting
Fun	O	O	O	O	O	O	O	Not fun
Appealing	O	O	O	O	O	O	O	Unappealing
Of no concern to me	O	O	O	O	O	O	O	Of concern to me

References

Ajzen, Icek. 1991. The theory of planned behavior. *Organ. Behavior and Human Decision Processes* 50(2) 179–211.

Alba, Joseph, John Lynch, Barton Weitz, Chris Janiszewski, Richard Lutz, Alan Sawyer, Stacy Wood. 1997. Interactive home shopping: Consumer, retailer, and manufacturer incentives to participate in electronic marketplaces. *J. Marketing* 61(3) 38–53.

Andrews, J. C., S. Durvasula. 1991. Suggestions for manipulating and measuring involvement in advertising message content. R. H. Holman and M. S. Solomon, eds. *Advances in Consumer Research*, vol. 18. Association for Consumer Research.

Anonymous. 1998. Repeat customers are scarce for most sites. *Internet World* 4(30) 14.

Assael, Henry. 1992. *Consumer Behavior and Marketing Action*. PWS-KENT, Boston, MA.

Atkinson, J. W. 1964. *An Introduction to Motivation*. Van Nostrand, Princeton, NJ.

Bagozzi, Richard P. 1980. *Causal Methods in Marketing*. John Wiley and Sons, New York.

Bakos, Yannis. 1997. Reducing buyer search costs: Implications for electronic marketplaces. *Management Sci.* 43(12) 1676–1692.

Bandura, A. 1982. Self-efficacy mechanism in human agency. *Amer. Psychologist* 37(2) 122–147.

Baty, J. B., R. M. Lee. 1995. Intershop: Enhancing the vendor/customer dialectic in electronic shopping. *J. Management Inform. Systems* 11(4) 9–31.

Beatty, Sharon E., M. Elizabeth Ferrell. 1998. Impulse buying: Modeling its precursors. *J. Retailing* 74(2) 169–191.

Bellman, Steven, Gerald L. Lohse, Eric J. Johnson. 1999. Predictors of online buying behavior. *Comm. ACM* 42(12) 32–38.

Berthon, Pierre, Leyland E. Pitt, Richard T. Watson. 1996. The World Wide Web as an advertising medium: Toward an understanding of conversion efficiency. *J. Advertising Res.* 36(1) 43–54.

Bhatnagar, Amit, Sanjog Misra, H. Raghav Rao. 2000. On risk, convenience, and Internet shopping behavior. *Comm. ACM* 43(11) 98–105.

Blakney, V. L., W. Sekely. 1994. Retail attributes: Influence on shopping mode choice behavior. *J. Managerial Issues* 6(1) 101–118.

Bloch, Peter H., Daniel Sherrell, Nancy Ridgway. 1986. Consumer search: An extended framework. *J. Consumer Res.* 13(1) 119–126.

Burke, Raymond R. 1997. Do you see what I see? The future of virtual shopping. *J. Acad. Marketing Sci.* 25(4) 352–360.

——, Bari A. Harlam, Barbara E. Kahn, Leonard M. Lodish. 1992. Comparing dynamic consumer choice in real and computer-simulated environments. *J. Consumer Res.* 19(1) 71–82.

Chatterjee, Samprit, Mark S. Handcock, Jeffrey S. Simonoff. 1995. *A Casebook for a First Course in Statistics and Data Analysis*. Wiley, New York.

Clawson, Pat. 1993. Study: Consumers want interactive TV. *Electronic Media* (August 23) 24–25.

Compeau, D. R., C. A. Higgins. 1995. Computer self-efficacy: Development of a measure and initial test. *MIS Quart.* 19(2) 189–211.

Cronbach, L. J. 1971. Test validation. R. L. Thorndike, ed. *Educational Measurement*. American Council on Eduaction, Washington, D.C.

Csikszentmihalyi, Mihaly. 1975. Play and intrinsic rewards. *Humanistic Psych.* 15 41–63.

——. 1977. *Beyond Boredom and Anxiety*. Jossey-Bass, San Fransisco, CA.

——, Isabella Selega Csikszentmihalyi. 1988. *Optimal Experience: Psychological Studies of Flow in Consciousness*. Cambridge University Press, Cambridge, U.K.

Davis, Fred D. 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quart.* 13(3) 319–340.

——, Richard P. Bagozzi, Paul R. Warshaw. 1989. User acceptance of computer technology: A comparison of two theoretical models. *Management Sci.* 35(8) 982–1003.

Draft, Howard C. 1993. The destiny of direct. *Direct Marketing* (December) 65–69.

Dykema, Evie Black, Kate Delhagen, Carrie Ardito. 1999. Consumers catch auction fever. *Forrester Res. Inc.*, Cambridge, MA.

Eighmey, John. 1997. Profiling user responses to commercial Web sites. *J. Advertising Res.* 37(3) 59–66.

——, Lola McCord. 1998. Adding value in the information age: Uses and gratifications of sites on the World Wide Web. *J. Bus. Res.* 41(3) 187–194.

Engel, James F., Roger D. Blackwell, Paul W. Miniard. 1990. *Consumer Behavior*. Dryden Press, Chicago, IL.

Fishbein, M., I. Ajzen. 1975. *Belief, Attitude, Intention, and Behavior: An Introduction to Theory and Research*. Addison-Wesley, Reading, MA.

Forman, Andrew M., Ven Sriram. 1991. The depersonalization of retailing: Its impact on the "lonely" consumer. *J. Retailing* 67(2) 226–243.

Gefen, David, Detmar Straub. 2000. The relative importance of perceived ease of use in IS adoption: A study of e-commerce adoption. *J. Association for Inform. Systems* 1(1) (http://jais.isworld.org).

Ghani, Jawaid A., Satish P. Deshpande. 1994. Task characteristics and the experience of optimal flow in human-computer interaction. *The J. Psych.* 128(4) 381–391.

——, Roberta Supnick, Pamela Rooney. 1991. The experience of flow in computer-mediated and in face-to-face groups. J. I. DeGross, I. Benbasat, G. DeSanctis, and C. M. Beath, eds. *Proc. 12th Internat. Conf. Inform. Systems*, New York, 229–237.

Greenwald, A. G., C. Leavitt. 1984. Audience involvement in advertising: Four levels. *J. Consumer Res.* 11(1) 581–592.

Heijden, Hans van der, Tibert Verhagen, Marcel Creemers. 2001. Pre-

dicting online purchase behavior: Replications and tests of competing models. *Proc. 34th Hawaii Internat. Conf. System Sci.*, Maui, HI.

Hoffman, Donna L., Thomas P. Novak. 1996. Marketing in hypermedia computer-mediated environments: Conceptual foundations. *J. Marketing* 60(3) 50–117.

——, ——. 1997. A new marketing paradigm for electronic commerce. *The Inform. Soc.* 13(January–March) 43–54.

——, ——, Patrali Chatterjee. 1995. Commercial scenarios for the Web: Opportunities and challenges. *J. Comput. Mediated Comm.* 1(3). ⟨http://www.ascusc.org/jcmc/⟩.

Hu, Paul J., Patrick Y. K. Chau, Olivia R. Liu Sheng, Kar Yan Tam. 1999. Examining the technology acceptance model using physician acceptance of telemedicine technology. *J. Management Inform. Systems* 16(2) 91–112.

Inman, J. J., L. McAlister, W. D. Hoyer. 1990. Promotion signal: Proxy for a price cut? *J. Consumer Res.* 17(June) 74–81.

Jarvenpaa, Sirkka L., Peter A. Todd. 1997a. Is there a future for retailing on the Internet? R. A. Peterson, ed. *Electronic Marketing and the Consumer.* Sage, Thousand Oaks, CA.

——, ——. 1997b. Consumer reactions to electronic shopping on the World Wide Web. *Internat. J. Electronic Commerce* 1(2) 59–88.

Korgaonkar, Pradeep K., Lori D. Wolin. 1999. A multivariate analysis of Web usage. *J. Advertising Res.* 39(2) 53–68.

Li, Hairong, Cheng Kuo, Martha G. Russel. 1999. The impact of perceived channel utilities, shopping orientations, and demographics on the consumer's online buying behavior. *J. Comput. Mediated Comm.* 5(2). ⟨http://www.ascusc.org/jcmc/⟩.

Lohse, Gerald L., Peter Spiller. 1998a. Quantifying the effect of user interface design features on cyberstore traffic and sales. *CHI*, Los Angeles, CA, 211–218.

——, ——. 1998b. Electronic shopping: Quantifying the effect of customer interfaces on traffic and sales. *Comm. ACM* 41(7) 81–86.

Maignan, Isabelle, Bryan A. Lukas. 1997. The nature and social uses of the Internet: A qualitative investigation. *J. Consumer Affairs* 31(2) 346–371.

Malone, Thomas W., Joanna Yates, Robert I. Benjamin. 1987. Electronic markets and electronic hierarchies: Effects of information technology on market structure and corporate strategies. *Comm. ACM* 30(6) 484–497.

Marakas, George M., Mun Y. Yi, Richard D. Johnson. 1998. The multilevel and multifaceted character of computer self-efficacy: Toward clarification of the construct and an integrative framework for research. *Inform. Systems Res.* 9(2) 126–163.

Mathieson, Kieran. 1991. Predicting user intentions: Comparing the technology acceptance model with the theory of planned behavior. *Inform. Systems Res.* 2(3) 173–191.

McQuarrie, Edward F., J. Michael Munson. 1992. A revised product involvement inventory: Improved usability and validity. J. F. Sherry and B. Sternthal, eds. *Advances in Consumer Research*, vol. 19. Association for Consumer Research, Provo, UT.

Mehrabian, Albert, James A. Russel. 1974. *An Approach to Environmental Psychology.* MIT Press, Cambridge, MA.

Miller, George A. 1956. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psych. Rev.* 63(March) 81–97.

Mitchell, A. A. 1981. The dimensions of advertising involvement. K. B. Monroe, ed. *Advances in Consumer Research*, Association for Consumer Research, Provo, UT.

Mittal, Banwari. 1989. A theoretical analysis of two recent measures of involvement. T. K. Srull, ed. *Advances in Consumer Research*, vol. 16. Association for Consumer Research, Provo, UT.

Morris, Betsy. 1987. As a favorite pastime, shopping ranks high with most Americans. *The Wall Street J.* July 30, p. 13.

Morrisette, Shelley, James L. McQuivey, Nicki Maraganore, Gordon Lanpher. 1999. Are Net shoppers loyal? *The Forrester Rep.* March.

Novak, Thomas P., Donna L. Hoffman, Yiu-Fai Yung. 1998. Modeling the structure of the flow experience among Web users. *Abstr. INFORMS Marketing Sci. Internet Mini-Conf.*, MIT.

——, ——, ——. 2000. Measuring the customer experience in online environments: A structural modeling approach. *Marketing Sci.* 19(1) 22–42.

Nowlis, Stephen M., Deborah Brown McCabe. 2000. Online vs. offline consumer decision making: The effect of the ability to physically touch merchandise. *Second Marketing Sci. Internet Conf.*, USC, Los Angeles, CA,

Nunnally, Jum C. 1967. *Psychometric Theory*, McGraw-Hill, New York.

O'Keefe, Robert, Tim McEachern. 1998. Web-based customer decision support systems. *Comm. ACM* 41(3) 71–78.

Park, C. W., Banwari Mittal. 1985. A theory of involvement in consumer behavior: Problems and issues. J. N. Sheth, ed. *Research in Consumer Behavior*, vol. I. JAI Press, Greenwich, CT.

Pedhazur, E., L. Pedhazur. 1991. *Measurement, Design, and Analysis: An Integrated Approach.* Lawrence Erlbaum and Associates, Hillsdale, NJ.

Peterson, Robert A., Sridhar Balasubramanian, Bart J. Bronnenberg. 1997. Exploring the implications of the Internet for consumer marketing. *J. Acad. Marketing Sci.* 25 329–346.

Pine, Joseph B., Don Peppers, Martha Rogers. 1995. Do you want to keep your customers forever? *Harvard Bus. Rev.* 73(2) 103–113.

Quelch, John A., Lisa R. Klein. 1996. The Internet and international marketing. *Sloan Management Rev.* 37(3) 60–75.

——, Hiroaka Takeuchi. 1981. Nonstore marketing: Fast track or slow? *Harvard Bus. Rev.* 59(4) 75–84.

Raman, Niranjan V. 1997. A qualitative investigation of Web-browsing behavior. *Adv. Consumer Res.* 24 511–516.

——, John D. Leckenby. 1995. Advertising in interactive media: Some issues, some suggestions. C. S. Madden, ed. *Proc. 1995 Conf. Amer. Acad. Advertising*, Waco, TX, 247–251.

Reichheld, Frederick F., W. Earl Sasser Jr. 1990. Zero defections: Quality comes to services. *Harvard Bus. Rev.* 68(5) 105–111.

Rice, Marshall. 1997. What makes users revisit a Web site? *Marketing News* 31(6) 12.

Rook, Dennis W. 1987. The buying impulse. *J. Consumer Res.* 14(September) 189–199.

Rotter, J. B. 1966. Generalized expectancies for internal versus external control of reinforcement. *Psych. Monographs: General and Appl.* 80(1, Whole N. 609).

Rowen, Mark. 1999. *Internet Retailing: The New Age of Retail.* Prudential Securities, New York.

Rust, Roland T., Richard W. Oliver. 1994. The death of advertising. *J. Advertising* 23(4) 71–78.

Shaw, T., S. L. Jarvenpaa. 1997. Process models in information systems. A. S. Lee, J. Liebenau, and J. L. DeGross, eds. *Information Systems and Qualitative Research.* Chapman and Hall, London, U.K., 70–100.

Shell, Adam. 1994. A map of the information highway. *Public Relations J.* 50(January) 27.

Sheth, Jagdish N., Rajendra S. Sisodia. 1997. Consumer behavior in the future. R. A. Peterson, ed. *Electronic Marketing and the Consumer.* Sage, Thousand Oaks, CA.

Slama, M. E., A. Tashchian. 1985. Selected socioeconomic and demographic characteristics associated with purchasing involvement. *J. Marketing* 49(Winter) 72–82.

Stern, Hawkins. 1962. The signifigance of impulse buying today. *J. Marketing* 26(April) 59–62.

Straub, Detmar, Richard T. Watson. 2001. Transformational issues in researching IS and Net-enabled organizations. *Inform. Systems Res.* 12(4) 337–345.

——, Moez Limayem, Elena Karahanna-Evaristo. 1995. Measuring system usage: Implications for IS theory and testing. *Management Sci.* 41(8) 1328–1342.

Szajna, Bernadette. 1996. Empirical evaluation of the revised technology acceptance model. *Management Sci.* 42(1) 85–92.

Tracy, Bernadette. 1998. e-Tailing: What Web customers really want. *Avertising Age's Bus. Marketing* 83(11) 39–41.

Trevino, Linda Klebe, Jane Webster. 1992. Flow in computer-mediated communication: Electronic mail and voice mail evaluation and impacts. *Comm. Res.* 19(5) 539–573.

Venkatesh, Viswanath. 1999. Creation of favorable user perceptions: Exploring the role of intrinsic motivation. *MIS Quart.* 23(2) 239–260.

——, Fred D. Davis. 1996. A model of the antecedents of perceived ease of use: Development and test. *Decision Sci.* 27(3) 451–481.

Webster, Jane, Linda Klebe Trevino, Lisa Ryan. 1993. The dimensionality and correlates of flow in human-computer interaction. *Comput. Human Behavior* 9(4) 411–426.

Xia, Lan, D. Sudharshan. 2000. An examination of the effects of cognitive interruptions on consumer online decision processes. *Second Marketing Sci. Internet Conf.*, USC, Los Angeles, CA.

Zaichkowsky, Judith Lynne. 1985. Measuring the involvement construct. *J. Consumer Res.* 12(3) 341–352.

Charles Steinfield, Special Issue Editor. This paper was received on January 10, 2000, and was with the author 10 months for 4 revisions.