

# A Design Theory for Pervasive Information Systems

Panos E. Kourouthanassis  
Athens University of Economics and Business  
Department of Management Science and Technology  
pkour@aueb.gr

## Abstract

Pervasive Information Systems (PIS) constitute an emerging class of Information Systems where Information Technology is gradually embedded in the physical environment, capable of accommodating user needs and wants when desired. PIS differ from Desktop Information Systems (DIS) in that they encompass a complex, dynamic environment composed of *multiple artefacts* instead of Personal Computers only, capable of *perceiving contextual information* instead of simple user input, and *supporting mobility* instead of stationary services. This paper aims at proposing a design theory for PIS. In particular, we have employed Walls et al. (1992) framework of Information Systems Design Theories (ISDT) to develop a set of prescriptions that guide the design of PIS instances. The design theory addresses both the design product and the design process by specifying four meta-requirements, nine meta-design considerations, and five design method considerations. The design theory has been validated through the implementation of an Information System in the retail context following the theory's prescriptions. A field experiment revealed that the design theory is capable of leading to valuable and acceptable PIS instances.

## 1 Introduction

Information technology (IT) artefacts are already embedded in more places than just our desktop computers, providing innovative services in ways unimaginable in the near past. This shift in the viewpoint of information systems (IS) is commonly referred to as 'post-desktop' (Jonsson, 2002) or 'ubiquitous' computing (Weiser, 2002). This trend has fired a shift away from computers towards computerised artefacts. A new generation of information appliances has emerged (Roussos, 2003), differing from traditional general-purpose computers in what they do and in the much smaller learning overhead they impose on the user. Instead of having IT in the foreground, triggered, manipulated, and used by humans, nowadays we witness that IT (irrespectively whether it comprises of computers, small sensors, or other

communication means) gradually resides in the background, monitoring the activities of humans, processing and communicating this information to other sources and intervening should it be required. This new class of IS is often called 'Pervasive Information Systems' (PIS) (Birnbau, 1997) and enables new interaction means beyond the traditional desktop paradigm.

This position paper aims at proposing a methodological approach that may facilitate designers to develop PIS instances. Specifically, we will present a design theory for PIS. The following section presents the design challenge of PIS. Section 3 briefly introduces the methodological framework that will be used to specify the design theory, while section 4 presents the theory itself. Section 5 is concerned with the activities we undertook to validate the proposed design theory. The final section concludes with a critical appraisal of the proposed design theory and its practical usefulness.

## 2 The Design Challenge of PIS

### 2.1 PIS Novel Characteristics

PIS introduce new elements in multiple dimensions spanning different IS domains, such as Human-Computer Interaction (HCI) and Software Engineering, which admonish us to examine them as a new Information Systems class. In essence, PIS revisit the way we interact with computers by introducing new input modalities and system capabilities. So far, the interaction paradigm for Information Systems has been the *desktop*. Thus, the design and implementation of Information Systems was based on this paradigm. PIS extend this paradigm by introducing a set of novel characteristics that are summarised in the following paragraphs.

First, PIS deal with non-traditional computing devices that merge seamlessly into the physical environment. As such, the desktop (in the form of the Personal Computer) is just 'another access device'. Consequently, conventional HCI design methods and interaction schemes may not be appropriate for the new IS class since the physical

interaction between users and the system will, most certainly, not resemble the prevailing DIS keyboard/mouse/display paradigm. On the contrary, PIS simulate the way that humans interact with the physical world. Abowd and Mynatt (2000) argue that since humans speak, gesture, and use writing utensils to communicate with other humans and alter physical artefacts, such actions can and should be used as explicit or implicit input to PIS. Burkey (2000) argues that the next step in this progression refers to environmental interfaces where the environment is the interface and the user exists in it. This is fully aligned with PIS where, ultimately, every artefact can interact with the system user. Thus, apart from solely physical interactions with the system, PIS may also incorporate elements of ambient interactions with devices or objects from the physical space (Schur, Decker, & May, 1999). According to the authors, “*these interactions should be lauded for their increased learnability and general ease of use*”. Additionally, they may be used by people with disabilities or IT unfamiliarity for whom the traditional mouse and keyboard are less accessible.

Moreover, PIS support a multitude of *heterogeneous device types* that differ in terms of size, shape (more diverse, ergonomic, and stylistic), and functionality (simple mobile phones, portable laptops, pagers, PDAs, sensors, and so on), providing continuous interaction which moves computing from a *localised tool* to a *constant presence*. Opposed to desktop environments where the access devices are stationary, PIS support *nomadic devices* which may be carried around by users and present *location-based information*. Since these devices are not required to be a fixed part of the pervasive system, PIS need to support *spontaneous networking*, implying ad-hoc detection and linking of the participating devices into a *temporary pervasive network* creating *dynamic dependencies* among the linked devices.

Furthermore, the participating elements of a pervasive system are *highly embedded* in the physical environment. This implies that they will inevitably interact with the existing architecture of the environment. Understanding architecture, therefore, plays a key role in designing PIS. Designers need to consider how architecture’s manipulation of space, aiming to minimise obstacles (Bentley, 1985), interacts with the pervasive artefacts that will eventually be deployed. Ultimately, architecture can be seen as a large-scale system, and as such, we can learn from its presence and design.

In addition, PIS emerge a revised viewpoint in the way we perceive system design. ‘Conventional’ system design incorporated more and more of the physical world inside the computer. In this sense, the actual system intelligence has been purely cybernetic, comprising of software designed to execute predefined tasks and activities efficiently. Moreover, systems were designed in such ways that enhanced overall utility and productivity, especially when applied to organisational contexts. In the case of PIS many computerised artefacts (instead of a single computer) monitor and support the user. The system’s intelligence no longer resides solely in the computer, but it is embedded in the physical world. Thus, each artefact may be specialised to support a *single task* performed in a more efficient way. This task may depend on a geographical location or may be triggered by an event such as a user request, a sensor reading change, and so on.

Building on the above, in desktop environments designers typically assume that user profiles are known in advance (Grudin, 1991a, , 1991b; Lynch & Gregor, 2004; Poltrock & Grudin, 1994), thus allowing for systematic requirements analysis. In PIS, the opposite may be true: it is highly unlikely for the system designer to know in advance the kinds of users who will be interacting with the system. Users may range from being vaguely familiar with IT to expert users. In addition, PIS users may be *opportunistic* in the sense that they may use the system only sporadically, implying that they may not be subject to training prior to system use.

Finally, PIS introduce the property of *context-awareness* as a result of the pervasive artefacts capability to collect, process, and manage environmental or user-related information on a real-time basis. Opposed to desktop computing, where *user action precedes system response*, PIS promote system *pro-action* based on *environmental stimuli*. This can be accomplished through the deployment of sensors and actuators in the physical world. The following table summarises the differences among the desktop paradigm (Desktop Information Systems – DIS) and PIS under six dimensions answering the following questions:

- What is the generic profile of a PIS user compared to DIS? (User)
- What interactions is the user expected to perform? (Task)
- How does the interaction take place? (Medium)
- Where does the interaction take place? (Space)
- What will be eventually designed? (Product)
- When will the system be used? (Time)

**Table 1:** DIS and PIS differentiating elements

	<b>Desktop Information Systems</b>	<b>Pervasive Information Systems</b>
<b>User</b>	<ul style="list-style-type: none"> <li>● Committed</li> <li>● Known</li> <li>● Trained</li> <li>Role Model: Office Clerk</li> </ul>	<ul style="list-style-type: none"> <li>● Opportunistic</li> <li>● Unknown</li> <li>● Untrained</li> <li>Role Model: Citizen</li> </ul>
<b>Task</b>	<ul style="list-style-type: none"> <li>● Specific</li> <li>● Focused on utility and productivity</li> </ul>	<ul style="list-style-type: none"> <li>● Generic</li> <li>● Focused on service delivery and experience</li> </ul>
<b>Medium</b>	<ul style="list-style-type: none"> <li>● Localised</li> <li>● Homogeneous</li> <li>● ‘Point and Click’ paradigm</li> </ul>	<ul style="list-style-type: none"> <li>● Constant Presence</li> <li>● Heterogeneous</li> <li>● Natural interaction and multimodal paradigm</li> </ul>
<b>Space</b>	<ul style="list-style-type: none"> <li>● Cybernetic</li> </ul>	<ul style="list-style-type: none"> <li>● Physical</li> </ul>
<b>Product</b>	<ul style="list-style-type: none"> <li>● Virtual</li> </ul>	<ul style="list-style-type: none"> <li>● Tangible and Virtual</li> </ul>
<b>Time</b>	<ul style="list-style-type: none"> <li>● Reactive</li> </ul>	<ul style="list-style-type: none"> <li>● Proactive</li> </ul>

## 2.2 The Need of a Design Theory for PIS

DIS, through their form of personal computers or other stationary access devices, were designed to fit into an office environment and the activities taking place there. They were designed to be efficient tools in the hands of professionals. Thus, their practice of interaction design is directed towards this setting. Obviously, everyday life is quite different from office work, and therefore other ‘places’, interfaces, and appearances have to be explored in order to find a broader repertoire of strategies for creating human-centered technology. Moreover, these systems are designed *for use*; this means that their design and evaluation are accomplished on the basis of some definition of their functionality and perceived usage. Thus, designers seek a solution that satisfies the basic criteria for usability such as efficiency in use, low error rate, and support for recovery from error, based on a general knowledge about what to do and what not to do to meet such criteria (Hackos & Redish, 1998; Nielsen, 1994). The objective is to achieve *maximum usability* with respect to a general, precise notion of use, and the design is motivated by this ambition.

Consequently, the existing design approaches for DIS follow the same rationale. Design methods such as SSADM (Weaver, Lambrou, & Walkley, 1998), ETHICS (Mumford & Weir, 1979), SSM (Checkland, 1981), or Object-Oriented Analysis (Mathiassen, Munk-Madsen, Nielsen, & Sage, 2000) to name but a few popular methods, consider systems that support predefined tasks and in many cases assume a job or office environment. Moreover, such methods rely on the knowledge of the designer in order to recognise potential problems and mostly offer a generic approach to design. Also, such methods tend to focus on details of systems, something which is possible with

traditional, static systems, but which cannot always happen for the dynamic and rich environments supported by pervasive systems. That is why the IS literature has classified such methods into broad categories each capable of tackling different problem classes. Finally, all the traditional approaches are oriented towards the fundamental tenets of Human-Computer Interaction: design for a specific user, performing a specific task, in a specific domain (Preece, Rogers, & Sharp, 2002). However, user interactivity with a pervasive system may not address these particular tenets.

It should be noted that traditional design approaches are still applicable for PIS environment. User participation in all its forms (user-centered design, participatory design, and so on) may still be employed to augment user acceptance of the final system. Structured design methods may be followed to develop pervasive systems. Still, they should be perceived under the prism of the novel PIS features. PIS are complex systems. This complexity may generate increased uncertainty to designers (e.g. whether they have collected and accommodated all system requirements). That is why current approaches to PIS design follow an iterative and incremental system implementation. Systems are designed as proof-of-concept prototypes that gradually evolve into final-cut systems. Emphasis is given to the successful deployment of technology and not on its actual use. Thus, priority is given to the resolution of engineering challenges, thus forcing designers to selectively omit from their design important system qualities such as usability, privacy management, and aesthetics.

This argument illustrates the value and utility of our research. A design theory may provide designers with a tool that holistically examines PIS and identifies, in a nut-shell, the key features that

should be addressed during the design process. Such a tool may influence the traditional design approaches and methods and suggest how these may be adapted to efficiently guide the design process in pervasive environments.

At the same time, the PIS domain itself requires the formulation of a comprehensive framework that supports designers throughout the design process. So far, PIS design has been stemmed from practice. It is a trial-and-error process. Thus, the knowledge generated so far presents only fragments of the PIS picture as a whole. Since PIS is a technology-driven phenomenon, some efforts that attempt to guide designers in a systematic way also perceive PIS design from a technical perspective. Others, view PIS design from a purely social perspective and emphasise on privacy management or environmental management issues. To summarise, there is no consolidated framework for PIS design. The absence of a common frame of reference forces most of current PIS implementations to be designed, deployed and evaluated in relative isolation. Indeed, current PIS implementations follow a vertical and ad-hoc approach implementing from scratch all the required elements based on the unique characteristics of the application domain.

### 3 Information Systems Design Theories

To specify our design theory we have followed the framework of Information Systems Design Theories (ISDT) that was first articulated by (Walls, Widmeyer, & El Sawy, 1992). An ISDT

aims at the design of classes of Information Systems, rather than the development of specific IS instances. To this end, design theories are prescriptive, in the sense that they provide constructs and guidelines for the achievement of stated goals, rather than explaining phenomena (explanatory theories) or predicting outcomes (predictive theories). Finally, design theories are theories of procedural rationality (Simon, 1996), as their objective is to prescribe both the properties that an artefact should have if it is to achieve certain goals, and the methods of artefact construction.

According to Walls et al. (1992), design theories have two aspects; one dealing with the product of design (the artefact that will form the outcome of applying the design theory) and another dealing with the process of design (i.e. the method by which the design product can be realised). We will use this distinction to describe the components that form an ISDT. These components are summarised in Table 2.

The framework has already been used to develop design theories for, amongst others, Executive Information Systems (EIS) (Walls, Widmeyer, & El Sawy, 1992), Decision Support Systems (Kasper, 1996), Group Decision Support Systems (Limayem, 1996), Organisational Memory Information Systems (Stein & Zwass, 1995), Simulation Systems for IS Evaluation (Giaglis, 1999), and Emergent Knowledge Processes (Markus, Majchrzak, & Gasser, 2002). The following section presents the proposed design theory for PIS.

**Table 2.** ISDT Components

<b>DESIGN PRODUCT</b>	
<i>Kernel Theories</i>	Theories from reference disciplines that govern design requirements
<i>Meta-Requirements</i>	The class of goals to which the theory applies
<i>Meta-Design</i>	A class of artefacts hypothesised to meet the meta-requirements
<i>Design Product Hypotheses</i>	Used to test whether the meta-design satisfies the meta-requirements
<b>DESIGN PROCESS</b>	
<i>Kernel Theories</i>	Theories from reference disciplines that govern the design process
<i>Design Method</i>	Description of procedures for artefact construction
<i>Design Process Hypotheses</i>	Used to test whether the design method results in an artefact consistent with the meta-design

## 4 Design Theory Components

### 4.1 Meta-Requirements Elicitation

The first meta-requirement refers to the profile of prospective PIS users. In the PIS context it is highly unlikely for the system designer to know in advance the types of people who will be using the system. Take into consideration the users of the different implementations of tour guides existing in the PIS literature (Abowd et al., 1997; Bederson, 1995; Bellotti, Berta, De Gloria, & Margarone, 2001; Davies, Cheverst, Mitchell, & Efrat, 2001); users may range from people that are vaguely familiar with IT (mainly due to their interaction with commonplace IT artefacts such as mobile phones) or (at extreme cases) techno-phobic. Moreover, these types of users are opportunistic in the sense that they will use the tour guides for a particular time frame and for a particular reason (in this example, to augment their visiting experience). To this end, it is highly unlikely that these users will be subject to thorough training in the system's use as in the case of DIS users. Conclusively, a PIS instance should support all the different user types by employing sufficient mechanisms that enhance or facilitate user interactions with the system, while at the same time haste users' learning curve for using the PIS.

The second meta-requirement refers to the PIS capability to support the multitude of different device types that may participate in the pervasive environment. Hansmann et al. (2003) distinguish among four types of devices: information access devices, intelligent appliances, smart controls, and entertainment systems. Nevertheless, ideally a PIS should support any device that has built-in active and passive intelligence. Therefore, devices' heterogeneity is the most important element that should be addressed during the design of a PIS. Moreover, the plurality and diversity of pervasive devices generate additional system requirements in terms of connectivity and integration between them (Saha & Mukherjee, 2003). Furthermore, the design of the application and user interface should take into account the unpredictability of end devices. In the case that an application follows the user and moves seamlessly between devices, it is implied that this application will have to adapt to changing hardware capabilities (different types of pointing devices, keyboards, network types, and so on) and variability in the available software services (Banavar & Bernstein, 2002).

Designing for manipulation of contextual information, implies that a PIS should be able to perceive relevant information of its environment (with location sensitivity and user identity capturing

being the minimum requirement as stated by (Abowd & Mynatt, 2000; Dey, 2001), process it, and adapt to changes in the environment taking into account both historical and current data. Although at present contextual information refers mainly to the users' current location, we expect that in the near future PIS will be able to perceive simultaneously multiple stimulants that may be contradictory one to the other. Thus, this meta-requirement suggests that PIS should accommodate an appropriate mechanism that will filter the different contextual information particles, process them, and adjust their behaviour according to the information that best suits the current occasion.

The final meta-requirement suggests that pervasive artefacts should be 'gracefully' embedded in the physical space. This smooth integration does not suggest that these IT artefacts should be completely invisible to the system users, as implied by most visionary research papers in the field (Norman, 1999; Satyanarayanan, 2001; Weiser, 1993, , 2002). On the contrary, we follow (Redstrom, 2001)'s considerations that pervasive technology should be governed by *meaningful presence*, promoting unobtrusiveness. Thus, the challenge is to design PIS in such a way that users perceive them as *part of the environment*. Universal design principles (Story, Mueller, & Mace, 1998) may be applied to create remembrances allowing for system usage with the minimal distraction. Likewise, the PIS designer should also focus on the aesthetical qualities of the pervasive artefacts (Djajadiningrat, Wensveen, Frens, & Overbeeke, 2004).

### 4.2 Meta-Design Elicitation

The first meta-requirement clearly suggests that PIS should support opportunistic or inexperienced users that do not have the luxury, or time, for training to the system's functionality. To this end, PIS designers should devise means that facilitate the interaction of such users with the system, and minimise its learning curve. The solution to this problem is to employ *natural, easy to use and easy to learn interfaces* that facilitate a richer variety of communications capabilities between humans and computation artefacts. At the same time, PIS designers should expect that the cases of PIS misuse will be increased compared to DIS. This is the result of the potential PIS users' profile (lack of experience and/or sufficient training).

Consequently, the system design should incorporate appropriate mechanisms that minimise the degree of errors, or guide the user in such a way that prevents errors from even occurring. Finally, PIS should be able to perceive users' current skill level through both the contextual information and current

system usage and adapt their functionality accordingly.

The diversity and plurality of pervasive devices poses new challenges for information delivery applications in this environment. To meet the demands in this heterogeneous environment, it is necessary for the information to be customized or tailored according to the user's preferences, client capabilities and network characteristics (Held, Buchholz, & Schill, 2002). As such, PIS should incorporate a sufficient adaptation system that can accommodate all different types of adaptations between different formats. Already several such mechanisms have been proposed in the literature (Berhe, Brunie, & Pierson, 2004; Gajos & Weld, 2004; Han et al., 1998; Lei & Georganas, 2001; Mohan, Smith, & Li, 1999; Ponnekanti, Lee, Fox, Hanrahan, & Winograd, 2001) providing the system designer with the option to select the most appropriate for its system requirements. Moreover, a PIS should be scalable. Scalability refers to the ability to incrementally increase the abilities of a system, whilst maintaining, or improving, performance (Saha & Mukherjee, 2003). As such, the issue of scalability refers mainly to ensuring smooth and unobtrusive communication among PIS clients and backend hardware infrastructure (such as sensors and actuators, backend systems, and so on).

Contextual management implies that sensing artefacts should be able to effectively communicate the information they collect and process as well as trigger events that deem necessary to support PIS users. Opposed to DIS where the user initiates the interaction with the system, the PIS vision for invisibility and unobtrusiveness suggests that the system is always active, continuously collecting contextual information, and pro-acting (rather than re-acting) to the needs and demands of the end users before they even start expressing them. Conclusively, the PIS designer should devise an appropriate mechanism that supports proactive system operation. Similarly, since it is extremely difficult to program each participating device and application in the pervasive environment to receive and communicate uniformly the information it collects, PIS designers should devise a representation format that is efficient enough to model, process and communicate context.

The final two meta-design considerations stem from the requirement of smoothly embedding the pervasive artefacts to the physical space. On the one hand, PIS should be easily accessible. If designers follow the extreme suggestion to completely hide the IT infrastructure, we might end up with a system that is completely inaccessible due to the fact that users are unaware of how to use it or, at

extreme cases, of its existence. On the other hand, pervasive artefacts should be smoothly embedded in the physical environment. As such, PIS designers should make sure that the systems which they create do not conflict with or challenge the architecture of the place they will be integrated. To reach that goal, co-operation between two previously completely different disciplines (namely IS software engineering and civil architecture) seems to be the logical solution. Consequently, the design of PIS should exploit the existing material of the physical environment and gracefully embed IT in the physical world.

#### 4.3 Design-Method Elicitation

The first design method consideration suggests that PIS designers orchestrate the design around the informal and unstructured activities that users perform. This is also illustrated in the functionality of various PIS implementations. For example, domestic PIS, in their multiple instantiations, support such activities as home automation (inventory management, light and heat adjustment, and so on), or home entertainment. The focus on activities, as opposed to tasks, is a crucial departure from traditional HCI design. Of course, activities and tasks are not unrelated to each other. Often an activity will comprise several tasks, but the activity itself is more than these component parts. The challenge in designing for activities is encompassing these tasks in an environment that supports continuous interaction.

The second design method consideration suggests that prototyping through its various forms (sketching, low-fidelity designing, and mock-ups, just to name a few) should be a core activity of the design process. Demonstrating prototypes to the system users will ensure the implementation of user-friendly interfaces, as well as the incorporation of user feedback regarding several dimensions of the system such as usability, functionality, privacy protection, and so on. Moreover, since PIS have significant impact to the physical environment (through the embedment of several pervasive artefacts), prototyping may assist designers to fix functional user requirements without implementing the PIS on a full-scale basis, thus, minimizing implementation costs in terms of time and financial resources.

The third design method consideration suggests integrating conceptual design in the early phases of the design process. Contextual design (CD) may be used in the form of sketching or conceptual scenarios to demonstrate the system's functionality to the system's stakeholders. Given that PIS may re-engineer the way users perform their tasks and

activities, CD may be employed as a technique that presents alternative usage scenarios, technological solutions, or interaction techniques, so that system users may evaluate them, and select the most user-friendly or appropriate, meeting their goals and aspirations.

The fourth design method consideration aims at addressing issues relating to user privacy. Indeed, context-awareness implies that the system will be able to monitor and process personal information such as the users' current location, activities, even information related to the human body (e.g. users' temperature, heartbeats, or respiration levels). Combining these features with the capability to store this information for future utilisation, it is not surprising that privacy protection is considered as one of the most major properties that a PIS should take into consideration from the very beginning of the design process with many researchers proposing specific guidelines or models that may be applied (Beckwith, 2003; Beresford & Stajano, 2003; Jacobs & Abowd, 2003; Langheinrich, 2001; Palen & Dourish, 2003).

Finally, the design of PIS should not be the concern of software engineers only. Software engineers have the necessary skills to analyse and design a system from an IS perspective: create entity-relationship diagrams, data-flows, large databases, select the most appropriate technical solution to the given problem, and so on. Therefore, they, most probably, lack the necessary skills to apply their design solution to the physical space on an effective manner, not to mention to propose the most effective solution based on the design problem. As such, the design team should be enriched with additional members that may involve types of people such as architects, or internal decorators, each providing a different perspective to the design of PIS. These people should be involved from the early stages of the design process in order to counsel software engineers on how to exploit environments' smart spaces. Likewise, they may advise them on how and where to place pervasive artefacts in an unobtrusive, aesthetical, and possibly invisible, to the system users', manner. Finally, they may recommend alternative layout propositions regarding the effective exploitation of the physical space in order to minimise hardware placement and environment restoration costs.

## 5 Theory Validation

Because ISDTs propose theoretical contributions (Walls, Widmeyer, & El Sawy, 1992, , 2004) we need to empirically test their propositions. Only the accumulated weight of empirical evidence will, in essence, establish the validity of any design theory

without any doubt. In our case, the proposed design theory has been employed for the design of a Pervasive Retail Information System (PRIS). Specifically, we followed the theory's prescriptions to design and implement a PIS capable of enhancing the shopping experience in supermarkets. In the supermarket environment, the shopper can pick up a wirelessly connected shopping cart equipped with a display device and a Radio Frequency Identification (RFID) sensor capable of scanning the contents of the cart.

The system functionality is summarized to the following scenario. The shopper could use her loyalty card to log in the system, which welcomed her and presented her the shopping list she had uploaded prior to her visit to the store. She could then start navigating within the store as usual, picking up products and placing them inside the shopping cart. Each time a product was placed in the cart, the display device showed its description and price, and updated the total cost of the cart contents. At the same time, the product was removed from the shopping list. Moreover, at any time the shopper could request additional information about a product (e.g. nutritional value or ingredients), get informed about ongoing promotional activities (fully personalised based on the shopper's profile and past consumption patterns), and request navigation assistance within the store. Finally, during checkout, the system transmitted the list of purchased products along with the total amount to the cashier, who billed the shopper and issued the receipt, thus alleviating the need for queues and product loading/unloading at the checkout. The detailed design of the pervasive system has been published in (Roussos et al., 2003) and (Kourouthanassis & Roussos, 2003).

To validate the propositions of the design theory we have generated a set of validation hypotheses measuring the instance's value and acceptance. To assess them, we organised a field experiment in a Greek supermarket (ATLANTIK) where the PRIS was used by supermarket shoppers to conduct part of their shopping. The field spanned over a two-week period. Sixty loyalty club members participated in the trial.

Initially, the trial environment was prepared in terms of (i) selecting the appropriate products, (ii) preparing a specially modified corridor inside the supermarket, (iii) preparing the technical infrastructure to support the trial, and (iv) inviting the trial participants. Participants were invited through phone interviews. The majority of the participants (85%) belonged to the age range of 30-54 years old, while 77% of the participants were female. The level of education was high, with over 71% of the participants having university or higher

education. Regarding familiarity with information technology, 66% were relatively familiar with PCs while 15% had never used personal computers before.

The execution of the trial was organized in a sequence of three distinct steps. Initially, the trial participants were shown the system functionality by a facilitator (10 minutes). After the end of the system demonstration, shoppers were prompted to use the system on their own (30 minutes). Shoppers were able to purchase the products that were displayed in the modified corridor. Following their interaction with the system, the participants completed a questionnaire evaluating the effect of the system on their shopping experience.

All research hypotheses have been validating which indicates that the proposed design theory results to valuable and acceptable PIS instances. In particular, subjects evaluated their shopping experience using the system as amusing and pleasant (with over 70% characterizing it as exciting), while 78% of them stated that the system enabled them to monitor effectively the products in their shopping cart while at the same time, organizing their supermarket purchases better. Furthermore, 85% of the participants stated that the system saved them time to search for additional information or promotional offers regarding the products they want to purchase.

Although the prototype was not fully integrated within the supermarket, shoppers identified that the use of such a system will eventually improve the check-out process, while almost all of them stated that they expect to wait less at the cashiers using such a system. An interesting observation derived from the fact that 89% of the trial participants stated that waiting less time in the cashiers would influence their decision to shop at a certain supermarket. Furthermore, shoppers perceived that the system improved the effect of time pressure within the supermarket since over 80% responded that the system offered them more time to conduct their shopping, while at the same time reduced the sense of time pressure and contributed to less hurry in the supermarket.

Regarding the continuous monitoring of the cart total value, 93% of the participants stated that the system can help them monitor their budget more effectively and that such a system allowed them not to spend more money than they have budgeted for. Finally, they responded that such systems may improve promotions effectiveness by presenting and organizing product promotions efficiently.

System acceptance was measured through the three major constructs of the Technology Acceptance Model (Davis, 1989) (perceived usefulness,

perceived ease of use, and perceived intention of use). The model was enhanced with two additional variables stemming from the design theory prescriptions (perceived privacy and perceived aesthetics). In all cases, the model fitted within acceptable parameters.

The detailed field experiment results concerning the system's value have been published in (Kourouthanassis, 2004). The field experiment results concerning the system's acceptance are available at (Kourouthanassis, 2006).

## 6 Conclusions

This position paper outlined the components of a design theory for the development of PIS. The theory consists of a set of meta-requirements, a set of meta-design considerations, and a set of design method considerations. This research stemmed from the lack of a consolidated framework to guide the design process of PIS.

We argue that the proposed design theory is an important theoretical contribution for several reasons that are summarised in the points below. First, the design theory *prescribes the design process and products of an emerging IS class*, which is expected to be the subject of significant investigation in coming years. This is evident by several publications in established IS journals, which advocate the significance of PIS in both practical and managerial terms, and specify agendas for further research in the field (Abowd & Mynatt, 2000; Lyytinen & Yoo, 2002; Lyytinen et al., 2004). Section 2 argued that this emerging field is in need of a systematic and holistic framework that consolidates the existing fragmented efforts regarding the design and development of PIS. Our work addresses this need by formulating a design theory for PIS, prescribing both the design product and the design process.

Second, the design theory provides *guidance to practitioners* by demonstrating how traditional IS design methods need to be modified to support the development of PIS. In most cases, traditional IS design approaches provide designers with little guidance about what to do and how to do it. In new and emerging IS areas (such as PIS), the existing knowledge base is often insufficient for design purposes and designers must rely on intuition, experience, and trial-and-error methods. The proposed PIS design theory collects the current knowledge and development practices in the PIS domain and presents them in a systematic and holistic way enhancing the efficacy of the design process. Similarly, in accordance to Hevner et al. (2004) call for effective communication of design

science outcomes, the PIS design theory facilitates managers *to conceptualise and understand the PIS phenomenon*. Our research identified the constituting components and critical factors affecting the deployment of pervasive technologies in a particular application domain, as well as their acceptance by users. Our work provides managers with a descriptive mechanism that helps them understand the PIS phenomenon, identify the capabilities and limitations of pervasive technologies, and recognise prospective business opportunities.

The most important limitation of our work refers to the validation of our theoretical propositions through a combination of single field experiment. We acknowledge that our design principles should be tested empirically in other situations where the same theoretical conditions hold; namely, PIS environments. Only the accumulated weight of empirical evidence will reinforce the validity of our theoretical propositions. We expect that our design theory will provide the initial grounds for further research in this area, collecting PIS common patterns, and specifying a set of generic design principles that may be followed by designers when developing PIS instances.

At the same time, our validation efforts focused on the utilitarian perceptions of the designed artefact for its users. We acknowledge that additional validation is required to address the perceptions of designers since they are the primary beneficiaries of this work. Although the design team that was involved in the prospective validation of the design theory provided us with several interesting insights, we believe that our theoretical propositions need to be exposed to a wider audience comprising of PIS designers that embrace a variety of design experiences and IS skills. It should be noted that an initial step towards this objective has been performed during the 3<sup>rd</sup> International Workshop on Ubiquitous Computing which was organised in Paphos, Cyprus during May 2006, where we presented our design theory principles. The workshop participants agreed, in principle, with our propositions and expressed the feeling that such an approach is required for the PIS field. Nevertheless, we posit that the PIS design theory needs to be the subject of additional debate and refinement in a more formal manner.

Finally, the prospective evaluation of the design theory did not address all of the theory's statements. Specifically, the design team did not address the principle concerning the adaptation of PIS functionality by identifying its users' current level of IT experience. Initially, the design team aspired to develop a mechanism that passively identifies the current level of IT experience based

on the pace and number of iterations that shoppers needed to scan the RF-tag, and the number of clicks in the system's available options. However, they decided not to develop this mechanism because the expected duration of each shopping session in the field experiment would be inadequate to properly test such a feature. The PIS would be used by each shopper only once since the supermarket's management did not provide us with the opportunity to conduct a longitudinal study. Moreover, each user would use the system for a time frame of approximately 30-45 minutes. As such, we realised that the system would not have the time required to adjust its services accordingly. Therefore, testing this element is left for future research.

The value of this research will be determined by the application of the proposed design theory by other scholars, in their effort to design PIS instances. We believe that this research provides an aggregated approach to describe PIS characteristics and prescribe their development. Moreover, it represents the only consolidated approach, to our knowledge, that investigates the problem of PIS design. In any case, the proposed design theory aims at specifying generic design principles that should be inherited by PIS instantiations. We decided to follow that paradigm in order to ensure that the proposed design prescriptions are applicable to all PIS instances, irrespective of their application domain. It is up to the PIS designers to interpret and apply these prescriptions based on their design problem.

## References

- Abowd, G. D., Atkeson, C. G., Hong, J., Long, S., Kooper, R., & Pinkerton, M. (1997). Cyberguide: A Mobile Context-aware Tour Guide. *Wireless Networks*, 3(5), 421-433.
- Abowd, G. D., & Mynatt, E. D. (2000). Charting Past, Present, and Future Research in Ubiquitous Computing. *ACM Transactions on Computer-Human Interaction*, 7(1), 29-58.
- Banavar, G., & Bernstein, A. (2002). Software Infrastructure and Design Challenges for Ubiquitous Computing Applications. *Communications of the ACM*, 45(12), 92-96.
- Beckwith, R. (2003). Designing for Ubiquity: The Perception of Privacy. *IEEE Pervasive Computing*, 2(2), 40-46.
- Bederson, B. B. (1995). Audio Augmented Reality: A Prototype Automated Tour Guide. Proceedings

- of the *ACM Conference on Human Factors in Computing (CHI '95)*, New York, 210-211.
- Bellotti, F., Berta, R., De Gloria, A., & Margarone, M. (2001). User Testing a Hypermedia Tour Guide. *IEEE Pervasive Computing*, 1(2), 33-41.
- Bentley, I. (1985). *Responsive Environments: A Manual for Designers*. London, UK: Architectural Press.
- Beresford, A. R., & Stajano, F. (2003). Location Privacy in Pervasive Computing. *IEEE Pervasive Computing*, 2(1), 46-55.
- Berhe, G., Brunie, L., & Pierson, J. M. (2004). Modeling Service-Based Multimedia Content Adaptation in Pervasive Computing. Proceedings of the *CF'04*, Ischia, Italy, April, 14-16.
- Burkey, C. (2000). Environmental Interfaces: HomeLab. Proceedings of the *Conference on Human Factors in Computing Systems*, The Hague, The Netherlands, 47-48.
- Checkland, P. (1981). *Systems Thinking, Systems Practice*. Chichester, UK: John Wiley and Sons.
- Davies, N., Cheverst, K., Mitchell, K., & Efrat, A. (2001). Using and Determining Location in a Context-Sensitive Tour Guide: The GUIDE Experience. *IEEE Computer*, 34(8), 35-41.
- Davis, F. D. (1989). Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly*, 13(3), 318-340.
- Dey, A. K. (2001). Understanding and Using Context. *Personal and Ubiquitous Computing*, 5, 4-7.
- Djajadiningrat, T., Wensveen, S., Frens, J., & Overbeeke, K. (2004). Tangible Products: Redressing the Balance Between Appearance and Action. *Personal and Ubiquitous Computing*, 8(5), 294-309.
- Gajos, K., & Weld, D. (2004). SUPPLE: Automatically Generating User Interfaces. Proceedings of the *Intelligent User Interfaces (IUI) 2004*, Funcha, Portugal.
- Giaglis, G. M. (1999). On the Integrated Design and Evaluation of Business Processes and Information Systems. *Communications of the AIS*, 2(5).
- Grudin, J. (1991a). Interactive Systems: Bridging the Gaps Between Developers and Users. *IEEE Computer*, 24(4), 59-69.
- Grudin, J. (1991b). Systematic Sources of Suboptimal Interface Design in Large Product Development Organizations. *Human-Computer Interaction*, 6(2), 47-196.
- Hackos, J., & Redish, J. (1998). *User and Task Analysis for Interface Design*. New York, NY: John Wiley and Sons.
- Han, R., Bhagwat, P., LaMaire, R., Mummert, T., Perret, V., & Rubas, J. (1998). Dynamic Adaptation in an Image Transcoding Proxy for Mobile WWW Browsing. *IEEE Personal Communications*, 5(6), 8-17.
- Hansmann, U., Merk, L., Nicklous, M. S., & Stober, T. (2003). *Pervasive Computing: The Mobile World (Second ed.)*. Berlin, Heidelberg: Springer-Verlag.
- Held, A., Buchholz, S., & Schill, A. (2002). Modeling of Context Information for Pervasive Computing Applications. Proceedings of the *6th World Multiconference on Systemics, Cybernetics and Informatics (SCI2002)*, Orlando, FL, USA, July, 14-18.
- Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly*, 28(1), 75-105.
- Jacobs, A. R., & Abowd, G. D. (2003). A Framework for Comparing Perspectives on Privacy and Pervasive Technologies. *IEEE Pervasive Computing*, 2(3), 78-84.
- Kasper, G. M. (1996). A theory of Decision Support System design for user calibration. *Information Systems Research*, 7(2), 215-232.
- Kourouthanassis, P. (2004). Can technology make shopping fun? *ECR Journal*, 3(2), 37-44.
- Kourouthanassis, P. (2006). *A Design Theory for Pervasive Information Systems*. Unpublished Dissertation, Athens University of Economics and Business, Athens, Greece.
- Kourouthanassis, P., & Roussos, G. (2003). Developing Consumer-Friendly Pervasive Retail Systems. *IEEE Pervasive Computing*, 2(2), 32-39.
- Langheinrich, M. (2001). Privacy by Design – Principles of Privacy-Aware Ubiquitous Systems. Proceedings of the *Ubicomp 2001*, Berlin, Heidelberg, 273-291.

- Lei, Z., & Georganas, N. D. (2001). Context-based Media Adaptation in Pervasive Computing. Proceedings of the *IEEE Canadian Conference on Electrical and Computer (CCECE'01)*, Toronto, Canada, May.
- Limayem, M. (1996). A design methodology for embedding decision guidance into GDSS. *Group Decision and Negotiation*, 5(2), 143-164.
- Lynch, T., & Gregor, S. (2004). User participation in decision support systems development: Influencing system outcomes. *European Journal of Information Systems*, 13(4), 286-301.
- Lyytinen, K., & Yoo, Y. (2002). Issues and Challenges in Ubiquitous Computing. *Communications of the ACM*, 45(12), 63-65.
- Lyytinen, K., Yoo, Y., Varshney, U., Ackerman, M. S., Davis, G., Avital, M., et al. (2004). Surfing the Next Wave: Design and Implementation Challenges of Ubiquitous Computing Environments. *Communications of the Association for Information Systems*, 13, 697-716.
- Markus, M. L., Majchrzak, A., & Gasser, L. (2002). A Design Theory for Systems that Support Emergent Knowledge Processes. *MIS Quarterly*, 26(3), 179-212.
- Mathiassen, L., Munk-Madsen, A., Nielsen, P. A., & Sage, J. (2000). Object Oriented Analysis and Design. Aalborg, Denmark: Marko Publishing.
- Mohan, R., Smith, J., & Li, C. S. (1999). Adapting Multimedia Internet Content For Universal Access. *IEEE Transactions on Multimedia*, 104-114.
- Mumford, E., & Weir, M. (1979). Computer Systems in Work Design - The ETHICS Method. London: Associated Business Press.
- Nielsen, J. (1994). Usability Engineering: Academic Press.
- Norman, D. A. (1999). The Invisible Computer: Why Good Products Can Fail, the Personal Computer Is So Complex, and Information Appliances Are the Solution: MIT Press.
- Palen, L., & Dourish, P. (2003). Unpacking "Privacy" for a Networked World. *CHI Letters*, 5, 129-136.
- Pollock, S. E., & Grudin, J. (1994). Organizational Obstacles to Interface Design and Development: Two Participant Observer Studies. *IEEE Transactions on Human-Computer Interaction*, 1(1), 54-80.
- Ponnekanti, S. R., Lee, B., Fox, A., Hanrahan, P., & Winograd, T. (2001). I-Crafter: A Service Framework for Ubiquitous Computing Environments. Proceedings of the *UBICOMP2001*, Georgia, Atlanta, September-October, 56-75.
- Preece, J., Rogers, Y., & Sharp, H. (2002). Interaction Design: Beyond Human-Computer Interaction. New York: Wiley and Sons.
- Redstrom, J. (2001). *Designing Everyday Computational Things*. Unpublished Doctoral Dissertation, Goteborg University, Goteborg, Sweden.
- Roussos, G., Spinellis, D., Kourouthanassis, P., Gryazin, E., Pryzbilski, M., Kalpogiannis, G., et al. (2003). Systems Architecture for Pervasive Retail. Proceedings of the *ACM SAC Track on E-Commerce*, 350-356.
- Saha, D., & Mukherjee, A. (2003). Pervasive Computing: A Paradigm for the 21st Century. *IEEE Computer*, March, 25-31.
- Satyanarayanan, M. (2001). Pervasive Computing: Visions and Challenges. *IEEE Personal Communications*, 10-17.
- Schur, A., Decker, S. D., & May, R. (1999). Design issues for next generation interfaces (NGI). Proceedings of the *Conference on Human Factors in Computing Systems (CHI'99)*, Pittsburgh, Pennsylvania, 130-131.
- Simon, H. A. (1996). The Sciences of the Artificial (3rd Edition ed.): MIT Press.
- Stein, E. W., & Zwass, V. (1995). Actualising organisational memory with Information Systems. *Information Systems Research*, 6(2), 85-117.
- Story, M. F., Mueller, J. L., & Mace, R. L. (1998). The Universal Design File: Designing for People of All Ages and Abilities: The Center for Universal Design, NC State University.
- Walls, J. G., Widmeyer, G. R., & El Sawy, O. A. (1992). Building an Information System design theory for vigilant Executive Information Systems. *Information Systems Research*, 3(1), 36-59.
- Walls, J. G., Widmeyer, G. R., & El Sawy, O. A. (2004). Assessing Information System Design Theory In Perspective: How Useful Was Our 1992 Initial Rendition? *Journal of Information Technology Theory and Application*, 6(2), 43-58.

Weaver, P. L., Lambrou, N., & Walkley, N. (1998). *Practical SSADM+*. London: Pitman.

Weiser, M. (1993). Some Computer Science Issues in Ubiquitous Computing. *Communications of the ACM*, 36(7), 75-84.

Weiser, M. (2002). The Computer of the 21st Century. *IEEE Pervasive Computing*, 1(1), 19-25.

