Part I

FRAMING THE PROBLEM AND CURRENT STATE-OF-THE-ART

You can put some informational part preamble text here. Illo principalmente su nos. Non message *occidental* anglo-romanic da. Debitas effortio simplificate sia se, auxiliar summarios da que, se avantiate publicationes via. Pan in terra summarios, capital interlingua se que. Al via multo esser specimen, campo responder que da. Le usate medical addresses pro, europa origine sanctificate nos se.

1 INTRODUCTION

> *The real problems going forward are not with any single device, but in the potential complexity of the larger ecosystem of technologies that we function in. [...] It's about the society of appliances and how they work today which is the new frontier.*
>
> — Bill Buxton [17]

Parts of this chapter appear in [58] and [59].

A key goal of ubiquitous computing [95] is "serendipitous interoperability", where devices which were not necessarily designed to work together (e.g. built for different purposes by different manufacturers at different times) should be able to discover each other's functionality and be able to make use of it [2]. Future ubiquitous computing scenarios involve hundreds of devices, appearing and disappearing as their owners carry them from one room or building to another. Therefore, standardising all the devices and usage scenarios a priori is an unmanageable task.

Next to serendipitous interoperability, another key goal of ubiquitous computing is to make technologies — as from a user's perspective they are still dealing with technologies — disappear, and "weave themselves into the fabric of everyday life until they are indistinguishable from it" [95]. To reach this goal, self-configuration of the various devices and technologies in ubiquitous computing environments is essential. Whether automated and initiated by context-aware entities, or initiated by users by connecting the devices to one another, the actual configuration of the various components at a lower level should happen automatically.

As computers disappear into the environment, we will need new kinds of human-computer interactions to deal with the peculiarities of these smart environments, which include invisible devices, implicit interaction, and distinguishing between physical and digital interactions [101]. In the conventional Graphical User Interface (GUI) genre, designers have typically developed prepackaged solutions for a predetermined interaction space, forcing users to adapt to their specific interaction protocols and sequences. In an ubiquitous computing environment, the interaction space is ill-defined, unpredictable and emerges opportunistically [22]. There is the risk of creating a mismatch between the system's model of interaction and the user's mental model of the system. In these conditions, new interaction techniques must be devised to help users to construct helpful mental models, in order to minimise system and user model mismatches.

A related issue is how ubiquitous computing differs from the sequential nature of traditional GUI interaction. The single point of control that is usually available in such interfaces naturally leads to a sequential organisation of interaction. One step inevitably leads to the next; as an example, consider a dialog box that refuses to let you do anything else until you click either `OK` or `Cancel`. When we interact with a smart environment, it is not only the parallel nature of the interaction with the physical world that is different, but also *the many different ways that we might map our tasks onto the features of the environment* [28]. Another difference is that these are not necessarily single-user interactions, but multiple users interacting in the same smart space at the same time.

If we are able connect smart devices to one another effortlessly, it becomes possible to support high-level services, that would usually involve multiple steps on multiple devices [72]. From a user's point of view, streaming music from a mobile device to a home entertainment system is a single high-level task. In practice there are multiple steps involved, and if the devices involved are from different manufacturers, the user needs to learn the operational details of each device interface in order to perform the task. Universal Plug and Play (UPnP) with its device control protocols [86] is not considered an adequate solution, because it only provides static device description documents and covers a very limited number of use cases.

At home the average person interacts with many devices during the course of a day. Sometimes these devices are used by more than one person, or one device may be used as an interface to another. As these devices are manufactured by different companies, there exist many different user interfaces that must be studied before they can be used. There might even be more than one way to interact with a single device. For example, to turn down the volume on a home entertainment system, either a remote control or a volume dial on the entertainment system itself may be used. It is expected that in future, more generic tools will be used to discover, configure, connect and control all the devices in the environment [54].

In a world where we are potentially surrounded by a multitude of devices, allowing for the arbitrary ad hoc interconnection of devices and the sharing of information between them is difficult. It is unreasonable to expect that a device will have prior knowledge of all the different ways it can interact with surrounding devices. The number of possibilities are too large, and anticipating the potential number of interactions is infeasible. If we could add meaning to the interactions and interconnections in such a way that it is machine-readable, semantic web technologies could be used to infer additional properties about the existing entities. This could fill the gaps between that which is described in terms of device capabilities, and that which is possible in terms of combined functionality. The user is still the final arbiter in deciding what the device does, but the device should be capable of communicating the possibilities based on what was inferred from its environment.

Besides the technological challenges, there also lies a challenge ahead for designing user interactions with these ecosystems of interconnected devices. When moving away from interaction with a single device towards interactions with systems of devices, designers need to find ways to communicate the relationships between the devices, and the larger system they are part of. Additionally, designers need to find ways to communicate the action possibilities of new, "emergent functionalities", that emerge when devices are being interconnected.

An important problem that arises when designing for these systems of interactive objects is their highly interactive and dynamic nature [35]. The inherent ever-changing nature of these systems and the severely limited overview of the ecosystem in its entirety is one of the most important challenges a designer faces when designing for such systems. Additionally, such a system comprises many different "nodes" that the designer, at the time of designing has no control over. Yet, when designing and adding new nodes to the system, making them interoperable is crucial for success.

According to Newman et al. [54], the following should be communicated to a user attempting to interact with and establish connections between devices:

- What devices and services are available
- Capabilities of the devices and services
- Relationships between each another and the environment
- Predictions of likely outcomes from interaction

The information presented to the user should be filtered dynamically, based on the user's context. This context includes the user's location, interaction history, and current tasks. A smart object is able to sense the context of its surroundings, make use of this context and other information to proactively interact with users and other smart objects, and self-organise into groups with other devices [74]. This context information should be represented in such a way that is understood by all the entities in the system.

1.1 DESIGN CHALLENGE AND RESEARCH QUESTIONS

This section describes the context of the work and the design challenge that was addressed. The work was completed in close collaboration with another PhD candidate, Bram van der Vlist, whose thesis [89] describes the more designer-related aspects in greater detail, whereas this thesis tends to focus on the more technical aspects of the work. Some overlap between the two theses is unavoidable, but we tried to keep this to a minimum.

The work described in this thesis was completed as part of a European research project called Smart Objects For Intelligent Applications (SOFIA)[1]. Some of the design choices were guided by collaboration with partners in the SOFIA project. We worked with the project partners to elicit requirements and expose ourselves to other application areas, in order to gain a more holistic view of the problem.

1.1.1 *The SOFIA project*

SOFIA is an European research project within the ARTEMIS framework that attempts to make information in the physical world available for smart services — connecting the physical world with the information world. The goal is to enable cross-industry interoperability and to create new user interaction and interface concepts, to enable users to benefit from smart environments. The centre of the software platform developed within SOFIA is a common, semantic-oriented store of information and device capabilities called a Semantic Information Broker (SIB). Various virtual and physical smart objects, termed Knowledge Processors (KPs), interact with one another through the SIB. The goal is that devices will be able to interact on a semantic level, utilising (potentially different) existing underlying services.

The SOFIA software architecture was taken as a departure point during each of the design iterations described in Part B. The various extensions and changes to the reference architecture are described in more detail in each design iteration description. A proposed software architecture to be used in future ubiquitous computing scenarios, based on the work done within SOFIA, is described in Chapter 10.

Our focus within the SOFIA project was on the user interaction aspects of devices in the smart home environment. While most of the examples in this thesis are specific to the smart home environment, the concepts are also applicable in the wider context of ubiquitous computing, for example the smart city or smart personal spaces.

1.1.2 *Ubiquitous computing*

Mark Weiser [95] coined the term ubiquitous computing, sometimes seen in its shortened form as ubicomp. It is describes a future where electronic devices are so ubiquitous that their presence is not noticed anymore. As described earlier in this chapter, we consider the key goals of ubiquitous computing to be serendipitous interoperability and making technologies disappear.

Chalmer and MacColl [20] questioned the more recent assumption in ubiquitous computing research that devices should disappear into the environment, reiterating Weiser's original vision that tools for in-

1 http://www.sofia-project.eu/

teraction should be "literally visible, effectively invisible". Devices should retain their unique characteristics, even when placed within systems of devices. Users are influenced by how they perceive devices, and we have to accept that the devices themselves are part of the user's context.

Ubiquitous computing products are a combination of hardware, software and services. It is not clear what kind of skills are required to design for this kind of environment [44]. There is, however, a need for interaction designers and software developers to have a common vocabulary and framework when cooperating to create these products. This thesis attempts to move this idea forward, by defining common concepts that are prevalent in most ubiquitous computing environments, and establishing a framework that can be used by both designers and developers alike.

1.1.3 *Ambient Intelligence*

The Ambient Intelligence paradigm differs from that of Ubiquitous Computing in that it tries to create environments that are sensitive to and responsive to the presence of people [3]. Devices disappear into the environment, necessitating virtual devices to support user interaction. It is expected that the devices adapt to to and even anticipate the user's needs. While our work is in principle closer to the vision of ubiquitous computing than ambient intelligence, there are some important aspects of ambient intelligence that need to be considered.

Marzano and Aarts [3] formulated the following five key technology features to define the notion of ambient intelligence:

- Embedded - many networked devices are integrated into the environment.
- Context aware - the system can recognise you and your situational context.
- Personalised - the system can tailor itself to meet your needs.
- Adaptive - it can change in response to you.
- Anticipatory - the system anticipates your desires without conscious mediation.

Personalisation refers to system adjustments made on short time scale, for example installing personalised settings. Adaption involves adjustments made by monitoring the user over a longer period of time. For anticipation, the system needs to be able to detect behavioural patterns that occur over a very long period of time.

We consider context awareness to be one of the most important features of a smart environment, especially when we consider a user's interaction with the smart space. Considering the parallel nature of our interaction with the physical world, any smart space will require context to help it make sense of the many different ways in which users map their tasks onto the environment.

Where the system tries to predict what the user is trying to accomplish, by being adaptive and anticipatory, we need to identify ways to give the users appropriate means to express themselves. The possibilities, available services and information that exist in the smart environment needs to be communicated in a meaningful way. Only if this is done correctly will users be able to build helpful mental models of the functionality the environment has to offer, set goals and make plans on how to act. By developing novel and meaningful interaction devices, the user can then perform the necessary actions and the system can in turn try to understand the user's goals and make the match to its internal models. We see a vital role here for the theory of *product semantics* [32], the study of how artefacts acquire their meaning and use its theories to define common concepts and semantics.

The ambient intelligence paradigm shows the importance of feedback in adaptive and anticipatory environments. In the thesis we describe the different kinds of feedback and feedforward that are applicable to these environments, as well as how it was implemented in a use case scenario.

1.1.4 *Affordances*

In their article "At Home with Ubiquitous Computing: Seven Challenges", Edwards and Grinter [30] describes a scenario where a couple come downstairs in the morning intent on listening to the radio, and realising that there is no sound coming from their speakers. It turns out that the neighbours bought a new set of Bluetooth-enabled speakers which, when installed, associated themselves with the nearest sound source – the couple's Bluetooth-enabled stereo.

The wireless nature of the speakers does away with the traditional affordances for understanding the connectivity between the speakers and the stereo, or even that the speakers can be connected to the stereo in the first place. These affordances are explicit when physical wires are used - the connections can be observed and the range of connectivity is clear. Edwards and Grinter state that the design challenge is to provide *affordances* that help users understand the technology, allowing them to control, use and debug technologies that interact with one another in the environment.

Norman [62] defined affordances as the set of possible actions of an object. An affordance is a relationship between an object and the person acting on the object, such that the same object might have different affordances for different individuals. The term was originally created by the psychologist J.J. Gibson to describe human perception, but was extended by Norman for its application to design.

Interaction metaphors [44] provide handles to these invisible technologies, where the metaphors are used to establish ideas about the meaning of physical affordances and potential ways to use these devices. For example, a Nintendo WiiMote is waved around much like a magic wand, so we can use "an enchanted object" as the interaction metaphor for the device.

1.1.5 *Ontologies*

The current state of ubiquitous computing is similar to that of desktop computing in the 1970s, where there is a whole range of new technologies without metaphors to communicate how they operate. The question then becomes how we then can model a device, not only in terms of its technical characteristics or capabilities, but from the user's point of of view, where metaphors, functionality and affordances play an important role.

One possible solution is to make use of ontologies, a concept in computer science most often associated with the Semantic Web [11]. Ontologies are formal representations of knowledge, consisting of various entities that are related to one another. They provide a shared vocabulary, which makes it easier to publish and share data. Ontologies allow us to model a domain in terms of its concepts, and the relationships between these concepts. They are also both machine-readable and human-understandable.

Ontologies are well suited to environments with a large number of devices. They have been designed to work at Web scale, they enable heterogeneous data sources to interoperate with one another, and they are based on technology standards which allow for easy and large scale adoption [74].

Ontologies lend themselves well for describing the characteristics of devices, the means to access such devices, and other technical constraints and requirements that affect incorporating a device into a smart environment [2]. Using an ontology also simplifies the process of integrating different device capability descriptions, as a semantic inferencing engine can be used to infer relationships between the concepts in the descriptions.

The hypothesis is that *user interaction in a smart environment can be better supported by ontological models than with existing device and service descriptions* (e.g. descriptions stored in relational databases). These ontological models define a semantic mapping between the user's behaviour and the available resources in the environment.

1.1.6 *Research questions*

"The greatest challenge to any thinker is stating the problem in a way that will allow a solution." – Bertrand Russell

The thesis aims to answer a number of research questions.

Research question 1. *In the previous section, ontologies were offered as a potential solution to solving the interoperability problem in ubiquitous environments. They are also well suited to describing user interaction in such an environment. This leads us to the question: How can we use an ontology to model user interaction and devices in a smart space consisting of multiple devices and multiple interactions?*

Research question 2. *How suitable is the publish/subscribe paradigm for handling ontology-based ubiquitous computing environments?*

How responsive is a networked user interface that is implemented on top of a system architecture with a semantic reasoning engine?

How can we measure the usability of ontologies and software frameworks for developers of ubiquitous computing environments?

How should feedback be provided in a networked user interface consisting of multiple connected devices?

1.2 OUTLINE OF THE THESIS

The following chapter describes related work, including relevant research projects. Existing state-of-the-art ontologies for ubiquitous computing environments and context-aware systems are described, followed by a description of the various interaction models, task models and semantic models that were used as basis for our own interaction model.

An iterative approach was followed during the design process. Part B describes the three design iterations, detailing the requirements, design, implementation and evaluation processes. A theory of semantic connections is introduced, based on the output from the design iterations, that focuses on the meaning of the connections between the different entities in a smart environment. It is intended to enable interaction designers and developers to create interoperable smart objects, providing them with a common vocabulary and framework.

Part C expands on the unique contributions of this thesis. Our approach to modelling the interaction capabilities of smart objects is described, which builds on earlier ontologies for context-aware systems. Another contribution of this thesis is in the way interaction events are modelled, utilising existing event modelling techniques to describe user interaction in smart environments. Ontology design patterns that were identified and used during the course of the design are documented. A software architecture based on the publish/-subscribe messaging paradigm is described, expanding on work performed within the SOFIA project. This is followed by an evaluation of the work, which includes a performance evaluation and usability analysis.

Part II

DESIGN ITERATIONS AND CONSTRUCTING A THEORY

You can put some informational part preamble text here. Illo principalmente su nos. Non message *occidental* anglo-romanic da. Debitas effortio simplificate sia se, auxiliar summarios da que, se avantiate publicationes via. Pan in terra summarios, capital interlingua se que. Al via multo esser specimen, campo responder que da. Le usate medical addresses pro, europa origine sanctificate nos se.

Part III

CONTRIBUTIONS AND EVALUATION

In this part of the thesis, the more general concepts and techniques that can be applied to ubiquitous computing are described. These concepts and techniques were extracted from work done during the three design iterations.

Part IV

APPENDIX