1. Introduction

"The idea of ubiquitous computing first arose from contemplating the place of today's computer in actual activities of everyday life".

Mark Weiser, "Some Computer Science Issues in Ubiquitous Computing," CACM, 1993

The mobile communication is stimulating the computing area for a new age. The possibility to use technological resources anywhere and at any time allows the users to perform their daily activities in both their professional and personal fields. However, it demands innovations in terms of architectures, models, new paradigms and other resources to support, for example, different user preferences, heterogeneous devices, distributed environments, several service providers, and accesses anywhere and at any time.

In order to contribute to this technological field, we propose a novel approach for the development of ubiquitous applications. Two main goals have driven our research: (i) the construction of reuse-oriented support sets based on an extensive analysis of ubiquitous applications and the Intentional-**M**ulti-**A**gent **S**ystems (MASs) paradigm – i.e. *Development for Reuse*; and (ii) the incremental and systematic development of Intentional-MAS-driven ubiquitous applications based on the reuse-oriented approach – i.e. *Development with Reuse*. Both research goals are complementary and our research addresses the possibility of reducing the technological gap between generic approaches – i.e. the traditional approaches that do not deal with specific ubiquitous concerns – and application-specific approaches – i.e. some approaches that are issue-specific-oriented, and are normally used to develop a unique application. On one hand, the former approaches demand a hard work to be specialized for ubiquitous applications. On the other hand, the latter approaches are really difficult to be reused.

Our proposal consists of an incremental and systematic development approach centered on intentional Multi-Agent-Systems (Pokahr et al. 2005) to improve the construction of ubiquitous applications. It involves the integration of distributed smart-spaces by using a specific platform and the wireless network as well as cognitive autonomous entities based on the **B**elief-**D**esire-**I**ntention (BDI) Model (Braubach et al. 2003; Bratman 1999). These two main mechanisms mean to incorporate to the developed ubiquitous applications some important properties, such as: autonomy, mobility, reactivity, proactivity, flexibility, adaptability, context awareness, among others.

The proposed technological support allows, for example, that each software agent gets information about the ubiquitous context under analysis without the need for external control, even if this context evolves or changes over time. Moreover, the support allows that in a ubiquitous environment, composed of different smart-spaces – i.e. it is structurally distributed – no centralized entity needs to have complete knowledge of the environment as a whole. The idea is to use cognitive entities working together to achieve specific goals in a collaborative manner. Furthermore, we also organized the proposed development approach in building blocks in order to improve the software engineers' work by promoting and facilitating the reuse of the available technological support sets from the requirements to code.

This Chapter presents in Section 1.1 the research questions. The contributions are discussed in Section 1.2. Finally, Section 1.3 describes the thesis outline.

1.1.
Research Questions

The first challenge for the development of ever-changing applications is that most of the research groups in the Software Engineering area focus their attention on the development of applications for desktops computers in client-server platforms. Therefore, it is important to adapt some concepts, methods, techniques and tools in order to deal with distributed computing, advanced networking resources and, more recently, the new concept of Ubiquitous Computing.

Even if we consider that there is plenty of interesting work about Ubiquitous Computing, it is not common to find contributions from the Software Engineering viewpoint to guide the software engineers in the development of ubiquitous applications from the requirements to code. In the literature, it is possible to find research publications in different areas (e.g. Artificial Intelligence and Distributed Systems) that propose solutions for specific ubiquitous concerns. However, it is difficult to find approaches that guide the software engineers in the systematic development of ubiquitous applications by emphasizing the importance of reuse-oriented support to deal with commonly found ubiquitous concerns.

To acquire a comprehensible view of ubiquitous concerns, we have investigated the literature and conducted extensive experimental work in the Software Engineering Laboratories at the **P**ontifícia **U**niversidade **C**atólica do **Ri**o de **J**aneiro (PUC-Rio) and of the **U**niversity **of T**oronto (UofT). Only to illustrate some ubiquitous concerns, let's describe a specific ubiquitous scenario centered on the patient's viewpoint from a dental clinic case study.

> *"I would like to find a dentist adequate to my dental problem, close to my actual location, who respects my privacy policies and whose price is compatible with my social conditions through the device I am using now. The request can be placed anywhere and at any time. I am a layperson in software technologies but I want to know what is going on and who is manipulating my personal data"*

As briefly presented, Ubiquitous Computing demands the use of heterogeneous and multi-functional devices, normally mobile, small and limited in terms of memory and processing capacity if we compare them with powerful desktop machines. Those devices use different types of networks, which are commonly limited in bandwidth. Moreover, the users of the ubiquitous applications are distributed, moving from one smart-space to another, with different preferences and desiring to balance personalization and privacy as well as invisibility and transparency issues. The personalization is important for the user's satisfaction. However, it demands a deep investigation of the user's preferences and other personal information, which can impact on the user's privacy. On one hand, the invisibility principle (Weiser 1991; Weiser and Brown 1995) idealized a world in which the services must be offered and daily activities must be performed to better attend the user, but without disturbing or even distracting her/him. On the other hand, the user also desires to know what is going on when a service is offered or an activity is performed – e.g. who is managing her/his personal data in the application (transparency issue). Furthermore, the ubiquitous environments, also known as ever-changing environments, are in constant evolution by following the commercial technological trends – e.g. novel devices with emergent technologies. Therefore, Ubiquitous Computing poses some challenges for the Software Engineering area, such as: device heterogeneity, distribution, user satisfaction, mobility, personalization *versus* privacy, invisibility *versus* transparency, and ever-changing environments. Our research questions were focused on ways to address these challenges. Summarizing these research questions, we can mention:

(i) *How can software engineers deal with the intentional modeling in ubiquitous applications?*

(ii) *How can software engineers deal with different quality criteria - from higher to lower abstraction levels - in ubiquitous applications?*

(iii) *How can software engineers deal with the device heterogeneity issue as well as the smart-spaces distribution in ubiquitous applications?*

(iv) *How can software engineers deal with the reasoning and learning of intentional agents in ubiquitous applications?*

(v) *How can software engineers improve the cognitive capacity of intentional agents in ubiquitous applications?*

(vi) *How can software engineers standardize the communication and improve the inter-operability of intentional agents in ubiquitous applications?*

(vii) *How can software engineers deal with the dynamic interface construction/adaptation in ubiquitous applications?*

(viii) *How can software engineers deal with specific ubiquitous issues (e.g. content adaptability and context awareness)?*

(ix) *How can software engineers deal with the intrinsic necessity of dynamic ubiquitous profiles' manipulation?*

(x) *How can software engineers deal with the evolution necessity in the device profile in order to follow the technological trends?*

In this field, there are different sub-areas in the Software Engineering that can contribute to improve the development of applications in Ubiquitous Computing by dealing with the previous research questions. In order to construct an adequate research-questions-based technological set to support the incremental and systematic development of ubiquitous applications, we investigated three main Software Engineering related areas: Software Reuse, Multi-Agent Systems (Zambonelli et al. 2000; Shoham and Leyton-Brown 2008; Bellifemine et al. 2007) and Goal-Orientation (Mylopoulos 2008; Lamsweerde 2001; Letier 2002).

One interesting area is Software Reuse. In this field, we take into account that a set of requirements based, for example, on some ubiquitous non-functional requirements (e.g. content adaptation, context awareness, distribution, mobility and device heterogeneity) are commonly found in different ubiquitous applications. Therefore, artifacts (e.g. conceptual models, frameworks, libraries and others) can also be constructed in order to be reused in the ubiquitous applications development. Furthermore, this computational support can compose a suitable technological set to facilitate and even to improve the development process based on reuse by reducing the time and the software engineers effort.

Other area that is gaining prominence is the Multi-Agent-Systems. Software agents have been applied to several contexts, such as: e-commerce, educational, network monitoring, and others. Therefore, we investigated, developed, evaluated, and now argue that the usage of software agents in Ubiquitous Computing can contribute to the development of ubiquitous applications. Some ubiquitous applications' cognitive domains that were used in our research to validate our ideas are the e-commerce and the dental clinic cognitive domains.

Moreover, we explored the application of some artificial intelligence techniques (e.g. reasoning techniques) to develop those agents by improving their cognitive capacity. In this field, we concentrated our attention on goal-oriented support. The Goal-Orientation paradigm has been one of the main topics of interest in different computing areas, especially in the Artificial Intelligence and the Requirements Engineering. The nature of this paradigm confers to the development of an application the possibility to be oriented by the users' goals and specific tasks to achieve these goals – i.e. their intentionality. Among other contributions and according to (Bratman 1999), the intentionality improves the like-me recognition (Gordon 2005), the human practical reasoning (Bratman 1999; Wooldridge 2000) and the goal formation (Dignum and Conte 1997). As our approach is providing resources for the development of ubiquitous applications from the requirements to code, it suggests the usage of the i* (iStar –

Distributed Intentionality) Framework (Yu 1997) in the requirements and design activities, and the **B**elief-**D**esire-**I**ntention (BDI) Model (Pokahr et al. 2005; Bratman 1999; Busetta et al. 2000; Georgeff et al. 1998) in lower abstraction levels. Both the i* Framework and the BDI Model are goal-oriented technologies.

Furthermore, the proposed incremental and systematic development mainly implies on: (i) the reduction of the documentation and the software engineers effort; and (ii) the reuse centered on models, frameworks, patterns and libraries, which are obtained from the *Domain Engineering of Ubiquitous Applications*.

1.2.
Contributions

Our research took place in the Software Engineering Laboratories at PUC-Rio and UofT under the supervision of Professor Carlos José Pereira de Lucena and collaboration of Professor John Mylopoulos. The **relevant contributions** of our work are:

- **Studies focused on the Requirements Engineering area.** In this field, we published two papers with a method to evaluate requirements teams: (i) in the *Workshop em Engenharia de Requisitos* (**WER'08**), the paper proposes the method by providing its detailed description; and (ii) in the *Simpósio Brasileiro de Engenharia de Software* (**SBES'08**), another paper reports on experiences acquired on the evaluation of that method.
- **The investigation of the state-of-the-art in different Software Engineering's transversal areas**, such as: Ubiquitous Computing, Intentional MAS, Goal-Orientation, and Software Reuse.
- **A Reuse-Oriented Architecture** centered on support sets – i.e. building blocks – obtained from the *Domain Engineering of Ubiquitous Applications*.
- **A Reuse-Oriented Application Engineering** for *Incremental and Systematic Development of Intentional Ubiquitous Applications* centered on the proposed building blocks.
- **Building blocks mainly composed of specific conceptual models, frameworks, patterns and libraries.** These building blocks are focused

on the main ubiquitous concerns (e.g. intentional modeling, distributed environments, content adaptability, mobility and context awareness) to improve the incremental and systematic development of intentional-MAS-driven ubiquitous applications in ever-changing contexts from the requirements to code. We published a paper in the *Proceedings of 11th International Conference on Enterprise Information Systems* (**ICEIS'09**), in which we describe some of these building blocks in the ubiquitous software development driven by agents' intentionality. Moreover, we published a paper in the *Proceedings of the 3rd International Conference on Agents and Artificial Intelligence* (**ICAART'11**) where we propose the use of intentional mobile agents to integrate ever-changing environments.

- **A reasoning engine focused on fuzzy conditional rules (Bigus and Bigus 2001) and the BDI model to improve the agents' cognitive capacity.** The fuzzy logic is centered on non-functional requirements (NFRs) (e.g. accountability, security, privacy, response time, and performance). These NFRs are commonly found in ubiquitous contexts and, according to our investigation, they are really relevant to the user satisfaction. We have an accepted paper – *Intentional Agents Supported by a Fuzzy-Logic-Based Reasoning Approach: An Application to a Ubiquitous System* – at *International Conference on Computer Science* (**ICCS'10/WASET**).
- **A specific mechanism based on intentional agents to deal with privacy issues in ubiquitous environments.** In this scenario, we focus our attention on the dependability, accountability, security, and their related quality criteria. Moreover, we also blend these criteria with other ubiquitous principles, such as: personalization, invisibility and transparency. The former one demands a deep investigation of the user's profile to acquire as much as possible her/his personal information in order to better satisfy her/him. However, it can contribute to the user's dissatisfaction as she/he sometimes does not want to share her/his personal information. Thus, it is relevant to balance personalization and privacy issues. The invisibility criterion desires a world that does not distract the user. In other words, every activity must be performed without disturbing the user. However, the user can desire to know what is going on.

Therefore, it is important to balance invisibility and transparency by allowing, for example, that the user knows who is manipulating her/his personal data – accountability issue. We submitted a paper to the *Journal Automated Software Engineering* (**AUSE**) proposing a development of privacy-aware ubiquitous applications using intentional software agents.

- **A NFR catalogue that graphically presents the main ubiquitous non-functional requirements, their interdependencies, and ways to operationalize them centered on the combination of traditional and emergent technologies.** In this field, we used the NFR Framework (Chung et al. 2000) and its **S**oftgoals **I**nterdependency **G**raphs (SIG) to model these requirements. Moreover, we opened this catalogue for our collaborators to facilitate the evolution of this catalogue based on different ubiquitous projects. Therefore, we organized the catalogue in a WEB application in order to facilitate its usage – e.g. consultation and instantiation – by third research groups. We published a book chapter *"Experiences with Requirements Model Reuse: The NFR Catalogue for Ubiquitous Systems"* in the *Handbook of Research on Mobile Software Engineering*, with the participation of Professor John Mylopoulos and Professor Eric Yu of the UofT.
- **Ontologies to allow the dynamic construction of interfaces and to improve the communication and inter-operability of the software agents in ubiquitous applications.** The interfaces are constructed by adapting them – at runtime – according to the ubiquitous profiles (e.g. user preferences, device features, and network specification) and centered on an ontological support. This ontological support, among other contributions: (i) specifies how an agent will present an interface form from the user's device; and (ii) standardizes the communication between the agents of the MAS platform. We published a paper in the *Proceedings of 12th International Conference on Enterprise Information Systems* (**ICEIS'10**), in which we present our results by applying the FIPA Standards Ontological Support (FIPA 2002a; FIPA 2002b; FIPA 2002c) to intentional-MAS-oriented ubiquitous systems. Moreover, we also published a paper in the *Proceedings of 1st International Conference on Pervasive and Embedded Computing and Communication Systems*

(**PECCS'11**) in order to present a dynamic interface adaptation for ubiquitous devices driven by agents and ontologies.

- **A dynamic database to store and retrieve the ubiquitous profiles (e.g. user, device, network and contract profiles) by improving the data management and the content adaptation "on the fly".** This database is based on a meta-architecture – called Type-Square (Yoder et al. 2001). This architecture allows the storage, retrieving, exclusion and other data manipulation at runtime by dynamically creating, modifying and deleting tables, attributes, and values. We published a paper – *Framework for Content Adaptation in Ubiquitous Computing Centered on Agents' Intentionality and Collaborative MAS* – in the *4th Workshop on Software Engineering for Agent-Oriented Systems* (**SEAS'08**). We also published a book chapter *"Dynamic Content Adaptation in Mobile Applications driven by Intentional Multi-Agent Systems"* in the *Handbook of Research on Mobile Software Engineering* by describing our dynamic content adaptation process for mobile applications based on the proposed dynamic database support. In addition, we submitted an extended paper to the *Journal Software: Practice and Experience* (**SPE**) with details of our dynamic content adaptation centered on intentional agents.

We intend to provide with the mentioned contributions a suitable engineering guideline with an appropriate technological set to help the construction of intentional ubiquitous applications by sharing good practices, reusing software and supporting their incremental and systematic development.

1.3. Thesis Outline

Here we present an overview of the thesis outline by summarizing each Chapter. Chapters 4, 5, and 6 are focused on the previously mentioned contributions.

Chapter 2 State-Of-The-Art This Chapter describes different transversal paradigms that are involved in our work. Therefore, we briefly present the Ubiquitous Computing, the Multi-Agent Systems, the Goal-Orientation and the Software Reuse paradigms. Moreover, we also discuss the combination of them.

Chapter 3 Related Work: Traditional & Emergent Approaches This Chapter provides some related work by focusing on traditional and emergent methodologies, approaches and processes proposed by the Software Engineering community. In particular, we present: RUP (Kroll and Kruchten 2003), TROPOS (Giunchiglia et al. 2003), GAIA (Román et al. 2002), Agile Methods (Benk et al. 2001), Mobile-D (Abrahamsson et al. 2004) and our reuse-oriented approach.

Chapter 4 Domain Engineering of Ubiquitous Applications This Chapter shows the technological support developed by considering different ubiquitous applications and their commonly found ubiquitous concerns, obtained from the *Domain Engineering of Ubiquitous Applications*. This support is reuse-oriented to facilitate the incremental and systematic development of ubiquitous applications. Moreover, it is based on emergent and traditional technologies. The i* Framework (Yu 1997) is mainly used to model intentional-MAS-driven ubiquitous applications by focusing on Goal-Oriented-Requirements Engineering (Mylopoulos 2008). The NFR Framework (Chung et al. 2000) is used to deal with non-functional ubiquitous requirements by allowing their modeling and the specification of their interdependencies and operationalizations. The JADE-LEAP Platform (Caire 2003) helps us to integrate the ubiquitous distributed environment as well as to deal with heterogeneous devices in ever-changing contexts. The JADEX Framework (Pokahr et al. 2005; Braubach et al. 2003; Braubach et al. 2004) provides resources – e.g. reasoning engine and capability concept (Braubach et al. 2005) – to implement intentional agents by using the BDI model. The Fuzzy-Logic Library (Bigus and Bigus 2001) improves the agents' reasoning to deal with non-functional requirements at runtime. The Type-Square Architecture (Yoder et al. 2001) is used to obtain a dynamic database model to manager data "on the fly". The WURFL Repository (WURFL 2010a; WURFL 2011b) is applied to dynamically evolve the device profile by following the technological trends. Finally, the Persistence Framework – e.g. Hibernate Framework (Hibernate 2011) – is used to manipulate the database. Therefore, this Chapter focuses on the *Development for Reuse*.

Chapter 5 Reuse-Oriented Approach for Incremental and Systematic Development of Intentional Ubiquitous Applications This Chapter presents the proposed incremental development for ubiquitous applications centered on intentional MAS. Here, we describe the reuse-oriented architecture by using the building blocks obtained in the *Domain Engineering of Ubiquitous Applications* (Chapter 4). These building blocks are used as they are provided and/or reused by instantiation or extension in the *Ubiquitous Application Engineering*. Therefore, this Chapter focuses on the *Development with Reuse* by also presenting a life-cycle with specific disciplines to guide/conduct the development process from the requirements to test.

Chapter 6 Our Proposal's Application This Chapter shows details of our proposal's application by using a dental clinic case study. Here, the focus is on the *Ubiquitous Application Engineering*, in which the support of the *Domain Engineering of Ubiquitous Applications* is reused in a specific case study.

Chapter 7 Our Proposal's Evaluation This Chapter is focused on the evaluation process by presenting the results obtained during our experimental research. Therefore, we describe the analyzed competences, the simulated environment, the participants' profile, the performed tests and the results' analysis by taking into consideration the dental clinic case study. We also present the dedication time and team effort for each discipline of the life-cycle based on this case study. Finally, we illustrate how our results influenced other research groups.

Chapter 8 Final Considerations This Chapter shows the final considerations by summarizing the thesis and highlighting its contributions and limitations. We also present some research problems suggested as future work.