HAL
open science

Self-describing objects with tangible data structures

Arnab Sinha

To cite this version:

Arnab Sinha. Self-describing objects with tangible data structures. Artificial Intelligence [cs.AI]. Université de Rennes, 2014. English. NNT : 2014REN1S024 . tel-01062441

HAL Id: tel-01062441

https://theses.hal.science/tel-01062441

Submitted on 9 Sep 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

N° d'ordre : 00000 ANNÉE 2014

THÈSE / UNIVERSITÉ DE RENNES 1
sous le sceau de l'Université Européenne de Bretagne

pour le grade de

DOCTEUR DE L'UNIVERSITÉ DE RENNES 1

Mention : Informatique

École doctorale Matisse

présentée par

Arnab SINHA

préparée à l'unité de recherche INRIA - Bretagne Atlantique
Institut National de Recherche en Informatique et Automatique
(Université de Rennes 1)

Self-Describing Objects with Tangible Data Structures

Thèse soutenue à Rennes le 28 mai 2014

devant le jury composé de :

Christine MORIN
Directrice de recherche, INRIA Rennes / *Présidente*

Nathalie MITTON
Chargé de recherche, INRIA, Centre Lille-Nord Europe et HDR / *Rapporteuse*

Pierre SENS
Professeur, Université de Paris 6 / *Rapporteur*

César VIHO
Professeur, Université de Rennes 1 / *Examinateur*

Charles CONSEL
Professeur, Université-IPB ENSEIRB-MATMECA, Bordeaux / *Examinateur*

Jean-Marc CHAMPEIL
Directeur de Projets techniques, Veolia Propreté / *Examinateur*

Michel BANÂTRE
Directeur de recherche, INRIA Rennes / *Directeur de thèse*

Paul COUDERC
Chargé de recherche, INRIA Rennes / *Co-directeur de thèse*

Dedication

This thesis is dedicated to the memory of my dear father, Anil Kumar Sinha. His constant guidance, quest for knowledge and passion for education has shaped me more than he could have ever known. He always encouraged me to increase my knowledge, and his last wish to continue my studies was a driving force in my decision to pursue a Doctoral degree. I wish you were here to witness this accomplishment. Keep showering your blessings upon me from wherever you are. With lots of love and "Pranam" to you, Papa.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my supervisors, Paul and Michel, for giving me the opportunity to research and complete my thesis within the team at INRIA. I'm thankful to them for constantly being supportive and attending to my concerns from Day One. During these three-plus years, most of my interactions were with Paul. Having insightful scientific discussions with him was always a pleasure, given his immense knowledge, experience, enthusiasm and last but not least, patience. On one hand, he always gave me the freedom to explore the area of my research, whilst on the other hand, guided me forward, warning about plausible pitfalls. What motivated me most was his holistic thinking towards solving research problems, even if it required proposing unconventional solutions. Outside academia, I discovered him to be enjoyable company while attending international conferences together. I could not have imagined having a better advisor and mentor for my PhD study.

My thanks goes to each one of the jury members, especially to Christine Morin for agreeing to be Chairperson. I also would like to thank Nathalie Mitton and Pierre Sens for accepting to review the manuscript and for providing valuable feedback.

This thesis was funded by the ANR (Agence Nationale pour la Recherche) as part of the Bin That Thinks initiative. My participation in this applied research project supplied me with invaluable learning experience, and my thanks goes to ANR and to all the members of teams from various disciplines that collaborated on the project. From this list, one colleague from my team, Yann Glouche, needs special mention for spending hours brainstorming on research problems with me.

I would also like to acknowledge all my friends and colleagues within the team, the organization and my social circles for your encouragement and support, especially during testing times. In spite of being far away from my family, you've all made my stay at Rennes quite enjoyable, both within the workplace and outside of it.

Finally, this list of acknowledgments would be incomplete without the special mention of my mother, sister, uncle and family members. This thesis would not have been possible but for their unconditional love, support and understanding, from half way across the globe. Talking over the phone with Arijit, my little nephew, was relieving during depressing times. Last but not least, I would like to acknowledge my friends back home for their continued friendship and support.

Contents

Acknowledgements 0

Table of Contents 0

Introduction 5

1 State of the Art 7
1.1 Pervasive Computing - its relationship to predecessors . . . 7
1.2 Interaction interfaces . . . 12
1.2.1 Tangible user interface . . . 13
1.2.2 Implicit user interface . . . 14
1.2.2.1 Virtual reality . . . 14
1.2.2.2 Ubiquitous computers . . . 15
1.2.2.3 Augmented interaction . . . 16
1.3 Perception with sensors . . . 16
1.4 Communication . . . 18
1.4.1 Mobile and telephony towards IP based services . . . 19
1.4.1.1 Second Generation Technology(2G) . . . 20
1.4.1.2 Third Generation Technology(3G) . . . 21
1.4.1.3 Fourth Generation(4G) . . . 21
1.4.2 WLAN for data communication . . . 22
1.4.3 Short range communication technology . . . 23
1.4.3.1 Bluetooth . . . 24
1.4.3.2 ZigBee . . . 25
1.4.3.3 Communication using the light spectrum . . . 25
1.4.3.4 Ultra Wideband (UWB) . . . 27
1.4.3.5 RFID . . . 27
1.5 Information architecture in Internet of Things (IoT) . . . 37
1.5.1 Tags data structure . . . 37
1.5.2 Data processing architectures . . . 38
1.6 Conclusion . . . 40

2 Smart Objects with Tangible Data 41
2.1 Smart Environments . . . 41
2.2 Self-Describing Objects . . . 42
2.2.1 Advantages . . . 43
2.2.1.1 Availability in absence network infrastructure and scalability 43
2.2.1.2 Interoperability . . . 44
2.2.1.3 Privacy and security . . . 45
2.2.2 The Tangible Data . . . 46
2.2.2.1 Scope of Tangible Data . . . 47
2.2.2.2 Applications using Tangible Data . . . 48
2.2.2.3 Challenges of Tangible Data . . . 49
2.2.3 Examples . . . 49
2.2.3.1 Selective sorting . . . 50
2.2.3.2 Detecting interactions . . . 51
2.3 Coupling Associated Objects . . . 55
2.3.1 Our Tangible Data Structure for describing related objects . . . 56
2.3.2 Vertex-labeled Graph . . . 60
2.3.2.1 The Smart Tool Box Application . . . 62
2.3.3 Distributed Tree Structure . . . 67
2.3.3.1 Data structure for hierarchical objects . . . 67
2.3.3.2 The Application . . . 71
2.4 Conclusion . . . 78

3 Tangible Data in Waste Management System 79
3.1 Bin That Thinks . . . 79
3.2 Waste flow and global architecture of the waste management system . . . 82
3.2.1 Wastes description . . . 82
3.2.2 Wastes identification . . . 83
3.2.2.1 Individual smart bins . . . 84
3.2.3 Cheaper alternative using QR codes . . . 84
3.2.4 Trash bag . . . 86
3.2.5 Collective container . . . 86
3.3 Prototype demonstration . . . 87

4 Conclusion 89
4.1 Perspectives . . . 90

A Smart Bin using OWL Ontology 95
A.1 Detailed System Design . . . 95
A.1.1 Input . . . 95
A.1.2 Knowledge Base (KB) . . . 95
A.1.3 Reasoning/Rules . . . 99
A.2 Applications . . . 99
A.3 Complexity . . . 100

B Publications 103
B.1 Conference Proceedings 103
B.2 Journal 103

C Résumé de la thèse en français 105
C.1 Introduction 105
C.2 Etat de l'art 107
C.3 Objets intelligents avec des données tangibles 110
C.4 Conclusion et Perspectives 112

Bibliography 128

Table of figures 129

Abstract 134

Abstract en français 136

Introduction

It has been quite sometime that the focus has shifted from the desktop computing model to the current trends of cloud or pervasive computing. Cloud computing has emerged as an important paradigm, widely adopted by enterprises. They possess enormous computing power and unlimited capabilities of data storage. The scalability of their processing capabilities can be supported on-demand basis. While the centralized infrastructure and operation of the cloud could be managed by third parties, it is possible for small organizations and individuals to use the cloud services remotely, anytime with an option to pay per-use basis. Pervasive computing, on the other hand, aims at integrating physical entities with the digital systems. The physical world is augmented by coupling digital information to everyday used physical objects and environment. This integration should be seamless and work invisibly remaining camouflaged within the environment. The physical activities and interactions influence and manipulate the associated information systems. The physical objects could be as trivial as waste items participating in pervasive applications to enable efficient sorting processes.

The cloud approach is very resourceful and flexible in accessibility. The classical Internet of Things envisions augmenting these cloud resources to limited capability mobiles and pervasive devices. The mobile and embedded devices also enable users to access these cloud-based services and data ubiquitously. Yet there are few challenging aspects to it. First, its architecture provides centralized storage. Hence the data needs to be accessed through network. This is a disadvantage as network connectivity might not be accessible from remote locations due to failures or security issues. Apart from dependency on the network, accessing every bit of data from the cloud would amount to overload on the infrastructure. Using the network has the problem of security. Among all the information transmitted over the network, at least the sensitive ones definitely needs to be secured by encryption. There is also the high risk of storing and analyzing the transmitted information while accessing cloud services. Furthermore, the data (specially the sensitive ones) stored in the cloud needs protection. They are always subjected to the risk of snooping. All said and done, the Big Brother is keeping a watch over activities.

Pervasive computing is an approach involving the physical world. Its physical entities within, are active participants of the computing process; since they are tightly coupled to the information system. The objects are piggybacked with information. When activities use these objects, their related information are used by applications in their computations. This is different from the previous, where the abstract computations take place on a remote cloud. We are following the pervasive trend, where information are in proximity to the physical objects.

In this doctoral thesis, we have studied an approach of storing data onto the physical objects using RFID. It is an emerging digital technology that is mostly used to link physical

objects to information systems. We have utilized an uncommon approach of RFID i.e. using their memory space for our purpose. It helps the objects provide self supported information system with advantages compared to the cloud approach. It has greater transparency when it concerns privacy. Deploying services using our approach requires minimal infrastructure (including network) which are performed locally. Hence, it is more scalable and dependable. But there exist some technological challenges too. Currently, RFID memory has a limited capacity. So, the data needs to be encoded efficiently within them. Our study proposes some efficient approaches. The semantic data stored in the objects provide the necessary context and information. All these required information would be available locally for pervasive applications to make inferences.

Our approach of coupled data enable to contain information pervasively within the physical environment. Hence, the data remains distributed in the surroundings within the proximity of the objects; contrary to cloud storage. As proof of concept, the approach was applied in the waste management domain. Waste objects are made smarter by using RFID tags containing self descriptive information. These objects can participate individually in waste sorting applications and some inferences depend on a collection of objects present locally.

Organization of this thesis

This thesis is split into four chapters. The first chapter provides a detailed overview of pervasive computing. It consists of multiple aspects like interaction, perception and communication; pointers to them are provided including their state of the art. RFIDs, which are considered to be a technology mostly used for short range communication, have been proposed to be used with a novel approach in this thesis. Hence, it is discussed in minute details to provide the background for our contribution. In the last section, we describe the various ways handling information for pervasive computing. This also forms an important section to differentiate between the various approaches of storing and retrieving the data, specially using RFID tags.

The second chapter presents the contribution of this thesis. It defends an approach where individual tagged objects utilize the digital memory to self-support information systems. The first demonstrates self-describing objects piggybacked with semantic information. Hence, relevant information associated with the physical entities are readily available locally for processing. The second establishes digital link among group of related objects using dedicated or ad-hoc data structure. This enables validating some property involving the objects in proximity. The advantages of our contributions are argued. They are also validated using demonstrator applications.

In the third chapter, we detail the fine aspects of “Bin That thinks”; a pervasive solution for smart waste management. It aims to perform efficient sorting and better recycling of self-described smart waste items. Finally, the concluding chapter summarizes the contributions of this thesis with some future perspectives.

Chapter 1

State of the Art

1.1 Pervasive Computing - its relationship to predecessors

Computer hardware and electronics make advancement at a very rapid pace. According to Moore's Law formulated in the sixties by Gordon Moore, the computing power or number of transistors on microchips doubles approximately 18 months. And until now the prediction has so far been quite true. In other words, it is stated as the form factor of computers have reduced significantly over the years. This resulted in quite a few changes. Firstly, it resulted in the increase in the ratio of computers per person and that has increased rapidly over the last few years. Secondly, devices became more portable and computers are being embedded with an increasing trend. In spite of all these, currently we are still a long way to go in achieving Mark Weiser's vision:

"The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it" [Wei91].

The remaining of this section would describe how the current scenario is related to the past systems. It also gives an intuition that the direction is right to achieve Mark's vision in the near future.

The author in paper [Sat01], describes how Pervasive computing is an evolutionary form of the work dating back to the mid-1970's. These predecessors were distributed systems and mobile computing whose problems have been extensively studied for solutions. While some of these problems including their solutions are applicable directly, others require advanced solution due to distinct constraints of pervasive computing. Moreover, pervasive computing has fresh problems to be solved due to its additional complexity. The relation between the evolutionary stages are drawn below.

A distributed system is defined as a collection of autonomous computers, connected through a network and distribution middleware, which enables computers to coordinate their activities and to share the resources of the system, so that users perceive the system as a single, integrated computing facility. These computers could be connected in a variety of ways; mobile or static, wired or wireless, sparse or pervasive. Its principle research areas could broadly be categorized

as: remote communication, fault tolerance, availability, remote information access and security [CDK01, Lyn96, Mul93]. These areas still concern pervasive computing. With the advent of laptops and wireless LANs were in the early 1990s, attempts were made to build distributed systems using these mobile clients. This effort emerged unique issues like network availability, client's energy and form factors, security and the new research direction called mobile computing. It is still an active research field and can be grouped into some broad research areas namely: mobile networking, mobile information access, support for adaptive applications, system-level energy saving techniques and location sensitivity.

Pervasive computing involves computing and communication technologies that should merge into the environment. It should have seamless interaction involving objects and users within its surroundings. Motion is an obvious activity for the users, if not the objects, in everyday life. Given such a situation, the authors in [Sat01] have put forth the statement:

"Since motion is an integral part of everyday life, such a technology must support mobility; otherwise, a user will be acutely aware of the technology by its absence when he moves. Hence, the research agenda of pervasive computing subsumes that of mobile computing, but goes much further."

The paper describes four additional concerns for pervasive computing. Firstly, the smart space (environment embedded with computing infrastructure) must be used as effectively as possible. Secondly, the technology should be invisible. The user interactions with technology should be experienced naturally and seamlessly. Thirdly, systems should be very adaptable to diverse smart spaces. The user experience of interaction should always remain constant to be unaware of the technological activities. The last concern for pervasive systems is scalability. With increased sophistication of smart spaces, the interactions between user's personal computing space with his surroundings would increase. This would have impact on bandwidth, computation and energy constraints. Localizing the computations would save data transmission bandwidth in addition to providing scalability. Moreover physical entities in proximity are liable to have more natural interactions with their relevant computations performed locally. Again, quoting the author in [Sat01] in this regard:

"Like the inverse square laws of nature, good system design has to achieve scalability by severely reducing interactions between distant entities. This directly contradicts the current ethos of the Internet, which many believe heralds the death of distance."

This thesis would try to address all these four concerns of pervasive computing discussed above. However, the focus would be more on the fourth aspect of localized scalability when it concerns some of the aspects like providing basic interactions to users, cutting down mostly on the network infrastructure.

Figure 1.1 shows how pervasive computing is related to mobile computing and distributed systems. Each research area has classical problems. Furthermore, newer problems emerge as well as the degree of complexity. The figure illustrates the new problems evolving moving from left to right whereas the modulation symbol indicates the multiplicative - rather than additive - nature of the problems.

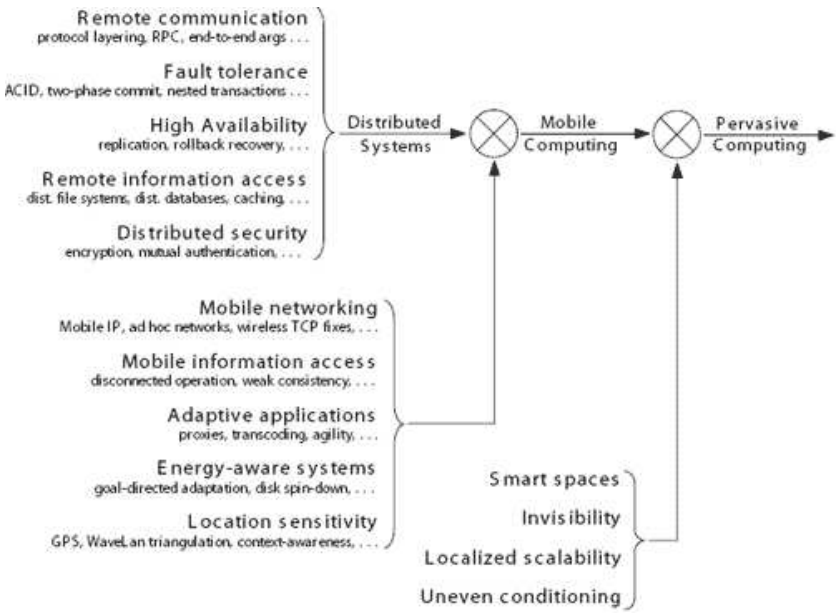

Figure 1.1: Evolution of Mobile and Pervasive Computing from Distributed Systems [Sat01]

General Model - Pervasive Computing

Pervasive Computing - envisioned by Mark Weiser - is the growing trend towards embedding processing power in everyday objects and environment so they can communicate information. The idea remains the same in the core; i.e. it is described as ubiquitous computing, ambient intelligence or everywhere, recently [Gre10]. Each of these emphasizes on slightly different aspects. Few other jargons such as physical computing, the Internet of Things [1], haptic technology [Sad07], and things that think [2] are also coined; however these are more penchant towards the objects involved [HPT97]. In short, each of them are named differently as varieties of the basic concept presented below.

The concept of pervasive computing is based on the idea of coupling the real world with information system. It serves two purposes: firstly, the activities are performed naturally in the physical space; secondly, the data processing within the information system are influenced directly by the physical activities. The approach towards developing pervasive systems is metaphorical. As the first step, a real world reference frame is chosen where the physical activities taking place would be considered. The activities are the actions involving objects and users in the reference frame. Consider for example, a living room as the frame of reference. The furnitures and equipments such as tables, chairs, sofa, entertainment system are the objects and the occupants, its users. Every object occupies a specific location in the frame. Some natural physical activities would be an object added or removed from the room. This

1. http://www.rfidjournal.com/articles/view?4986
2. http://ttt.media.mit.edu/vision/vision.html

physical world requires coupling to the information system, to perform pervasive computing. Abstracting the physical space to an array in the virtual space would be the simplest way. And each of its element would uniquely correspond to an object; like storing the UPC [3] (Universal Product Code) bar code for each object. Physical activities like addition or removal of objects can be equated to insertion or deletion of the corresponding element from the array; thus, data processing within the information system as a consequence of physical activities. Another activity like looking for a physical object would be congruent to searching for the element in the data structure; i.e. array in our case. Various types of abstractions, exploiting information system concepts are often used for representing real world digitally, inspired by analogies. Some examples are:

- using data structures like arrays, list, tables, heaps, graphs etc.
- concepts of memory space and addressing
- conceptualizing computing processes as activities
- abstracting synchronizing activities with semaphores or monitors
- interprocess communication as a way of expression between activities

Abstracting the physical world within information space sets the basic framework to perform pervasive computing. The physical reference frame and its entities within have a virtual representation and hence they bear a dual existence. However for the concept to be working, tight integration of both these environments is necessary. Computer systems must acquire the status of the real environment, and be able to act accordingly. This is called context awareness (context -awareness). Referring to the smart living room, discussed earlier, physical activities like adding or removing piece(s) of furniture changes the context of the environment. This must be reflected in the information system as well. Apart from objects, context awareness also includes user activities in its ambit. This is an advancement unlike the traditional approach where the users had to interact, explicitly with computer systems [Wei93].

Figure 1.2 shows the three main elements of pervasive computing along with an illustration, explained chronologically as follows. First, the physical environment which contains the real-world entities (objects, people, places, etc.). The second is the virtual or digital environment, consisting of information systems, such as the Internet. The third group is a set of interfaces that allows the coupling between the two environments.

The information system knows the state of the real world by sensing the environment (c). It could be the position of a person (acquired through a positioning system such as GPS), weather information of a place (acquired by specialized sensors), the operational status of a device etc. The survey of the environment therefore covers all the techniques to automatically update the digital information on events or real-world entities. In contrast, interfaces can act on the real world through the digital environment (d). For example, the windows of a building or a car may possibly be automatically closed when it starts to rain.

In a pervasive computing environment, the user accesses the information space implicitly (b), by its actions in the real environment. This implies that computing is available (either dedicated for the user or shared nearby), that is capable of enriching the environment with useful information as well as ease its interaction with the real world. This technical aspect of pervasive computing called augmented reality [FMS93a, FMHW97, NR96, RN95]. Pervasive

3. http://www.gs1us.org/resources/standards/ean-upc

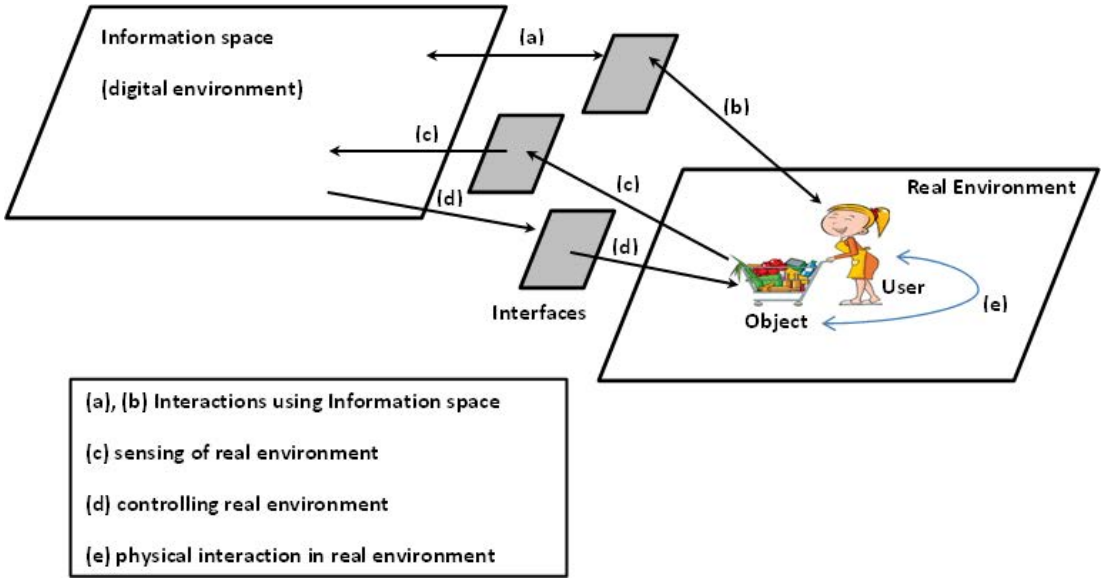

Figure 1.2: General Model - Pervasive Computing

computing requires the use of wireless communication to interface multiple objects with the information space. Using wired communication for connecting these likely mobile objects is unfeasible [Wei93]. On the other hand, the information space can probe the real environment, through interfaces; implicitly (a). Thus reducing on the voluntary actions required by the user to provide information.

Figure 1.3: Smart trolley with interactive features

As an illustration in figure 1.2, shows a buyer shopping in a pervasive supermarket. On entering, Diana grabs a smart trolley - like figure 1.3 - which asks to identify herself using fidelity card. The cart has a display for interaction with her; mostly displaying information to aid her shopping experience. She starts shopping around with the cart and acquiring items she needs. This is physical activities occurring naturally in the real environment (e). Additionally, the carts are smart to sense the items put onto them; total bill and weight is updated automatically and displayed (a & b). The supermarket has wireless infrastructure to locate and

communicate with carts (c). They also have her past shopping history. When she is located with her cart at certain aisles, the display pops up suggestions; based on either Diana's past or general shopping trend (d). Some of the recommendations made to Diana are:

– you're shopping vegetables. don't forget milk, that you always take.
– you just picked up apples! generally you take oranges too. They are located adjacently.
– there's cheese cake mix in the cart. probably you would need some eggs located nearby.

Finally, during checkout, Diana would just spend a few seconds to make the payment; since bill is pre-calculated.

The general model presents a very high level view of pervasive computing working. It aims to unify two different worlds - real and virtual - such that physical activities continue to occur naturally, while, the latter takes a back seat. Although numerous approaches are proposed by researchers, the basic building blocks are information space, real environment and interaction interface. Technology is used to build and/or connect these blocks; nevertheless, it must remain invisible to the users. Each of these components are emerging research areas. Their conceptual overview are discussed in the remaining sections below, along with discussions on their state-of-the-art.

1.2 Interaction interfaces

It has been decades since the computers have been invented to process digital information. Its hardware technology has made drastic progress over the years which has reduced its form factor by several order; previously, mainframes occupying an entire building to embedded processors sized few millimeters. Yet the interaction with the digital information is mostly confined to the Windows, Icons, Menus, Pointing device (WIMP) interface, also referred as Graphical User Interfaces (GUI). GUIs have been in existence since the 70's and first appeared commercially in the Xerox 8010 Star Information System in 1981 [SIKH82]. It offered bit-mapped display interface, one step forward than its predecessor, the Command User Interface (CUI) which was text based.

Although GUIs have been around for decades, researchers have acknowledged the requirement for better ones. [RN95] points out its fundamental limitations:

– Explicit operations : Although GUI conceals the knowledge of operating system commands and microprocessor instructions, the steps by and large needs to be performed graphically. In this way, users adopting a computer for completing tasks efficiently tends to get distracted from the activity at hand. The human-computer interaction becomes more engaging than the rather important human-real world interaction. Consider for example a user drawing a picture or typing a document with an image editor or word processor. He could spend considerable amount of time looking for menus and buttons, if the program interface is not well acquainted to him. People would want to be less bothered with the computer operations while performing a real world task; like painting or writing.
– Unaware of real world situations : Although laptops, mobile devices or PDAs are portable in nature, their GUIs are mostly static. These devices are meant to be carried around and used in a variety of situations. But unfortunately, their traditional GUIs do

not adapt to the context of its environment.

– Gaps between the computer world and the real world : The "see, point and click" interaction of GUI is far better than CUI that requires users to"remember and type" characters. Yet, there exists a gap with the real world. When a user wants to access or update digital information, he must somehow "quit" the real world interactions and focus on the computer [See figure 1.4(a)]. Although recent GUIs - which are quite advanced - facilitate interactions with computers, yet the user is required to handle highly remote abstractions of the real world. This "isolation" is acceptable for a fixed workstation, because the user can concentrate fully on their activity on it. The situation is quite different when we consider progressing towards Weiser's ubiquitous computing having usage of embedded computers potentially everywhere and at any time. The interaction with the virtual environment should be as natural as possible, if not invisible or implied. Consider a situation, a passenger needs access to flight information arriving at the airport, on his mobile phone. Presently, although technically possible, this scenario is not realistic considering the amount and complexity of required man-machine interactions. The passenger, typically pressed and stressed, should begin by seeking the connection to the Web, find and enter the website address of the company operating the flight or airport operator, and finally enter the reference of the flight; all with the interface necessarily limited to a portable device, like a mobile phone.

Hence we see that the current traditional and graphical peek-through-the-window into the digital world does not fit perfectly for pervasive computing. In the past two decades, there have been various approaches proposed with aspiration to merge the physical and digital world seamlessly. Some of these proposed interfaces, engage the human senses for their active participation. To name a few: touch-based interface, gesture-based interface, voice-based interface and tangible user interface (TUI); the latter has seen increasing interest among Human Computer Interaction (HCI) researchers [ZGO13]. Apart, there are some HCI styles of implicit or multimodal user interfaces where computer assists the interaction.

1.2.1 Tangible user interface

This area of research is receiving more and more attention these days and coming up with novel techniques for tangible devices. Also, they are truly towards ubiquitous computing. TUI has evolved out of the seminal work presented as "Graspable User Interface" by Fitzmaurice et al. [FIB95]. They had demonstrated a concept where physical artifacts named "bricks" (by them), allows direct control of electronic or virtual objects. Their work proved three principle ideas:

– the "bricks" act as handles for control
– it leverages user's comfort of manipulating physical objects
– the Graspable UI design offers a space multiplexed design with a one to one mapping between control and controller

Two years later, Ishii and Ullmer put forward another perspective of HCI. They explained that we, human beings, live between two realms of cyberspace and physical environment; i.e. bits and atoms respectively [IU97, Ish08]. They proposed to reduce this divide and coupling digital information to everyday physical objects and environment. They coined the term "Tangible

User Interface". In the paper, they presented a series of prototypes - metaDESK, transBOARD and ambientROOM - to support three of their concepts of Tangible Bits; interactive surfaces, coupling of bits and atoms, and ambient media. According to Ishii's experience, "... giving tangible (physical) representation to the digital information, TUI makes information directly graspable and manipulable with haptic feedback" [Ish08].

Eventually, TUI prototypes were proposed for variety of application domains like learning, programming, problem solving and entertainment. Some of the prominent ones among them are: AlgoBlocks [SK95], Digital Manipulatives [RMB+98], Electronic Duplo Blocks [WP02], SystemBlocks [ZR03], Topobo [RPI04], Boda blocks [BE07], Tangible Programming [Hor08], Media Blocks [UI99], Block Jam [NDNG03] and many more. Some HCI researchers have studied and classified TUI systems [HB06]. Finally, TUI prototypes are not only limited to the research domain but also have been commercialized; Sifteo [4] and Topobo [5].

1.2.2 Implicit user interface

In this section, we provide a brief overview of the existing HCI flavors (illustrated using figure 1.4). It would enable the reader to draw a contrast between them. As pointed out in the earlier section, GUI (Graphical User Interface) is still the most traditional and common form of interaction used today [see figure 1.4(a)]; i.e using a monitor, mouse, keyboard etc. by human users to interact with the digital world.

1.2.2.1 Virtual reality

A solution to integrate real environment and the digital environment is virtual reality [in figure 1.4(b)]; the user is immersed in a virtual environment that simulates a real environment. Every sense of the user (vision, hearing, touch , etc..) is connected to devices converting real signals describing the virtual world in the digital environment. This solution is very suitable for some applications (games, military etc.. Simulations), cannot however be used as a universal approach. Its high cost of computing power for simulating the exhaustive real world makes it infeasible. Secondly, it is not even desirable, as it would result in a total isolation of individuals from the real world.

Augmented reality (AR) is a variant of virtual reality that could use see-through head mounted displays to overlay computer generated images on the user's real sight [BFO92, FMHS93, FMS92, FMS93b, FS91, Sut68, RN95]. The authors in [NR96] have demonstrated two augmented reality based experimental systems - ShopNavi and WalkNavi. In the former, the user wears a small voice-activated computer, equipped with positional and orientation sensors and a wireless tag reader. The customer sees on his screen, commercial information guidance about the product he looks at and navigational help in shops. WalkNavi is a similar device helping pedestrians to navigate; it retrieves location related information from Geographic WWW Server [NR96]. Voice control system allows for more natural interactions than devices like keyboard, mouse, touchscreen with stylus etc. Summing up, it could have varied application domains across the real world spectrum.

4. Sifteo http://www.sifteo.com
5. Topobo http://www.topobo.com

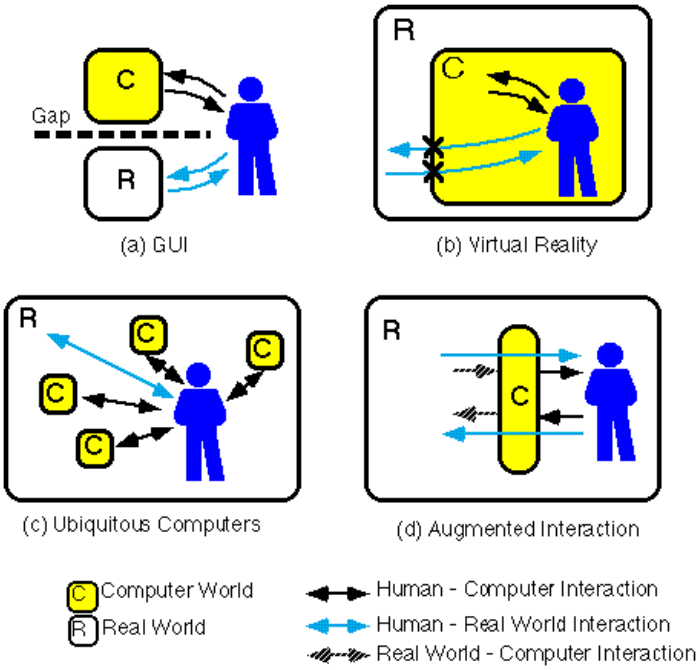

Figure 1.4: A comparison of HCI styles [RN95]

1.2.2.2 Ubiquitous computers

The second approach [figure 1.4(c)] is described by Weiser [Wei93]. The real world is enriched with a large number of specialized computers, embedded in real everyday objects (books, lamp, coffee maker, air conditioning, etc..). The user can continue to interact normally with the real environment, which also has been enriched through on-board computer interactions. It has a small computer which provides a kind of window to the digital part of the environment. The computer is used each time in a specific context for explicit interactions with user interfaces. Implicit interactions with the digital environment are more transparent and traditional. Consider the example of a library in a ubiquitous computing space as per Weiser's approach. The user has a laptop to find a reference book. He remembers the context where the book was recommended; like an e-mail, Website , or an electronic "post-it" provided by a colleague met at a restaurant. We already see here that the user memory is "extended" artificially by computer. Once the reference is selected, the mobile computer can contact via a wireless network server library to locate the book and its related ones. Then, instead of presenting the user the raw data, it is possible to guide him directly to the relevant documents in the real world. Fitzmaurice proposes a prototype in which the LCD strips are placed along the shelves with books, to highlight books that match a search [Fit93]. The user can further press the LCD strip in a book that interests him, to display the table of contents, for example.

The cost of these solutions is prohibitive. Also, it would be ambitious to expect that the cost will drop to a point, where the environment could be covered with such a high density of

computers. However, such approaches allow for seamless integration of the digital environment in the physical environment without isolating the user, which can also continue to interact normally with the real environment, if desired.

1.2.2.3 Augmented interaction

This approach [as in figure 1.4(d)] combines with the previous virtual reality techniques [figure 1.4(b)], to further increase the transparency of interactions. Indeed, in the prototypes as ParcTab [SAG+93] and Chameleon [Fit93], the user is aware of manipulating a computer. Although interactions between the user and computer is simplified, yet it has real knowledge of the environment. So, the experience is similar to working on a workstation. The idea of augmented reality [Fit93] or enhanced interaction [RN95] is to extend the natural abilities of the user to facilitate its interaction with the physical environment. For this purpose, its approach borrows visualization techniques directly from virtual reality and require a relatively complete survey of the situation of the user (position, orientation of the head or eyes and possibly biometric parameters such as heart rate, body temperature, etc..).

In augmented interaction, the user is aware of changes in the real environment; unlike virtual reality, where the picture is projected to the user, enriched by the information acquired in the digital environment. This approach is particularly appropriate for example like maintenance of complex devices (such as a laser printer) [Fit93]. The operator wears a helmet with camera that provides real time instructions to perform operations for maintenance. Another prototype allows a person to move into a campus with an enriched view of the environment, such as indications that appear on buildings observed [FMHW97]. Its use has been demonstrated in the automotive domain to ease driving [SABH09].

Compared to ubiquitous computing (a high-density environment of computers), the “augmented reality” approach is advantageous; in terms of cost-effectiveness. Instead of integration machines in each object, an embedded system for the user with a wireless network access could serve many purposes. However, interactions with the actual environment are less direct, since the user is partially isolated by the equipment. It could be felt uncomfortable. Additionally, since they use a logical representation of reality, consistency must be ensured between this representation and the physical world. This step won’t be necessary when objects are “smart”, itself.

1.3 Perception with sensors

The previous section presented the various interfaces by which computers interact with the real world, which includes all living and non-living things. This research underwent phases of changes led by factors such as reduced size but increased efficiency of processors, shift in the people-computing power ratio, changes in people’s activities of daily living etc. While this interaction research aims to minimize the gap that distinguishes between the real and virtual environments, it focuses mostly on the proactive activities. However the physical world is full of activities occurring continuously. They are presented through an incredibly rich set of input modalities, including acoustics, image, motion, vibration, heat, light, moisture pressure, ultrasound, radio, magnetic, and many more exotic modes [ECPS02]. The activities need to be

perceived through these modalities. Instrumenting the physical world with pervasive network of sensors would enable to capture the activities which are not proactive, in a more realistic manner. All the information gathered is represented, stored, reasoned and actuated in the information world using the ubiquitous and invisible computers present in the environment.

A variety of environmental sensors are available for perceiving a physical space. There are sensors for detecting motion, position measurement, for detection of chemicals and humidity sensing, to determine readings for light, radiation, temperature, sound, strain, pressure, position, velocity and direction. Ambient intelligence (AmI) is an emerging research area that aims to make our everyday environment intelligent and sensitive to us. The European Commission first chartered a path of AmI research in 2001 [6] which combines the use of environmental sensors and their network, pervasive computing, and artificial intelligence [DBS+01]. For example, Yatani and Truong has demonstrated in their prototype "BodyScope", that record the sounds produced in the user's throat area while performing different activities like eating, drinking, speaking, laughing and coughing. The first for activities could be identified in a small scale study - at 71.5% accuracy - due to the richness of information in their acoustic signals [YT12].

Physiological sensors are of different variety that can measure vital parameters like body and skin temperature, heart sounds and rate, respiration rate, activity, mobility, fall, electrocardiograms (ECGs), electroencephalograms (EEGs), electromyograms (EMGs), blood glucose, or SpO_2. These biosensors are being used to develop smart wearable systems (SWS) for health monitoring (HM) [CEF+12]. These sensing systems have various form factors such that they can be:

- worn by an individual as an accessory [JDCP10, RBR+07, AWL+04, LGO+10, WCH+09, HDT+08, Ma11, MSM07, LSDR10, BBS+08, Cou07, SSL+07]
- implantable, in vivo [BM01, MPF09, CMS10, FFHR07, HMWC12, PJA+10, AOH07, CMD11]
- portable
- embedded in the user's outfit as part of clothing [GTL+09]
- Embedded in pieces of object, furniture, or the floor of a house [SSVL+02, CAJ09, CP08]

Smartphone

This section until this point discussed about various kinds of sensors that are available and used for different purposes. Some are deployed and distributed in the pervasive environment for perception while others are assembled to build devices like SWSs. Modern smartphones are a unique kind of device that are used for both purposes. Basically a smartphone is a mobile phone built on a mobile operating system, with more advanced computing capability and connectivity than a feature phone. They also have powerful embedded sensors such as accelerometer, digital compass (magnetometer), gyroscope, global positioning system (GPS), microphone, proximity sensor, ambient light sensor (luxmeter), and camera; see fig 1.5. The sensors enable new sensing applications to be developed across a wide variety of domains such as social networks,

6. http://www.ist.hu/doctar/fp5/istagscenarios2010.pdf

mobile health, gaming, entertainment, education and transportation.

Figure 1.5: Smartphone sensors

There are applications and prototypes that gather data using smartphone's sensors distributed across physical locations. For example, Liu has proposed a novel real-time traffic information collection system which recognizes the driving status and route using accelerometer, compass and GPS of the user's smartphone. This data for traffic condition is updated onto the server [LWW+13].

There are millions of applications where smartphones act as intelligent devices making use of its sensors. You et. al. have demonstrated a soon-to-be-released Android application that uses the front-facing and rear camera on a dashboard-mounted smartphone to detect dangerous driving behaviors and driver fatigue [YLC+13, YMdOB+12]. It detects abrupt lane changes and when a driver is following another car too closely, and it monitors the driver's eye blinking rate to gauge fatigue. Audible and visual alarms provide early warning to help prevent accidents. Another application "StressSense" by Lu et. al., recognizes stress from human voice using smartphones [LFR+12]. It simply makes use of its embedded microphone achieving an accuracy of 81% and 76% for indoor and outdoor environments, respectively. [DTB+10] detects and alerts dangerous vehicle maneuvers using accelerometer and orientation sensor of the mobile phone placed in vehicle.

1.4 Communication

We are familiar with the current telecommunication and broadcasting systems (both wired and wireless) that provide various services for voice, data and television. They have been enabled by the digital convergence based on IP based framework. It is not only human operators but also machines who benefit using these communication infrastructures. Embedded devices

communicate among themselves (called M2M communication) for providing ambient intelligence. Additionally, communication technologies also act as an enabler for perceiving the physical environment. This last aspect is discussed at the end of this section.

Ubiquitous and pervasive systems are made up of embedded computers and sensors in the environment. The omnipresent sensors perceive the physical situation. Their captured data needs to be accumulated and assimilated for complete awareness. Hence data communication is necessary to process the information cooperatively.

For instance, sensors are placed in a combat zone for monitoring and spying. Figure 1.6 presents the situation pictorially. The sensors monitor the environmental conditions such as pressure, temperature, vibration, sound etc. They can communicate among themselves to perform the monitoring task more efficiently as well as to transmit information to army's remote base stations.

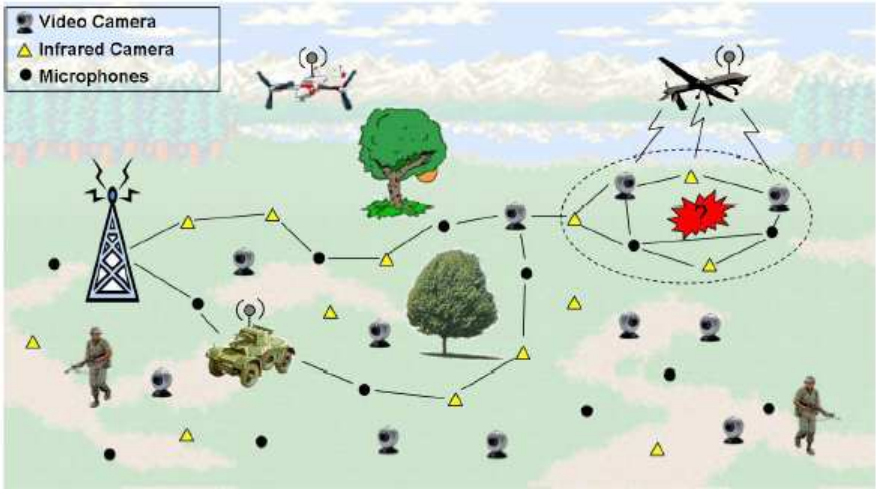

Figure 1.6: Sensor Networks in battlefield

There are various types of wireless networks used for pervasive services. Their features vary based on the type of infrastructure, the network range, frequencies used, the type of signal modulation to increase channel efficiency and channel sharing, the bandwidth available and power consumption. All these networks have various pros and cons and hence they are used suitably according to the requirements.

1.4.1 Mobile and telephony towards IP based services

Mobile communication has evolved manifold in the last few decades and has pervaded through the life of human beings. In fact, its heavy impact has changed their lifestyle. Currently we have just stepped into the Fourth Generation (4G) of wireless access technology. The cellular generation (G) refers to a change in the fundamental nature of the service, non-backward compatible transmission technology and new frequency bands. The technologies have evolved depending on the requirements. The First Generation (1G) of mobile commu-

nications were the analog telecommunication standards that were introduced in the 1980s. It was meant to carry analog voice signals and used Frequency Modulation (FM) and Frequency Division Multiple Access (FDMA). Eventually with the growing success and demand, it kept improving with their characteristics and offered newer features. The next three subsections would discuss briefly its later generations which are capable of data transmission; hence used for M2M communication. Table 1.1 provides an quick overview comparing all the generations of mobile technology [Sha13].

Technology Features	**1G**	**2G**	**3G**	**4G**	**5G**
Start/ Deployment	1970-1980	1990-2004	2004-2010	Now	Soon(probably 2020)
Data Bandwidth	2kbps	64kbps	2Mbps	1Gbps	Higher than 1Gbps
Technology	Analog Cellular Technology	Digital Cellular Technology	CDMA 2000 (1XRTT, EVDO) UMTS, EDGE	Wi-Max LTE Wi-Fi	WWWW (coming soon)
Service	Mobile Telephony (Voice)	Digital voice, SMS, Higher capacity packetized data M2M	Integrated high quality audio, video and data M2M	Dynamic Information access, Wearable devices M2M	Dynamic Information access. Wearable devices with AI capabilities M2M
Multiplexing	FDMA	TDMA, CDMA	CDMA	CDMA	CDMA
Switching	Circuit	Circuit, Packet	Circuit, Packet	All Packet	All Packet
Core Network	PSTN	PSTN	Packet N/W	Internet	Internet
Coverage Area	from tens to thousands of KMs				

Table 1.1: Comparison of all Generations of Mobile Technologies [Sha13]

1.4.1.1 Second Generation Technology(2G)

The 2G era of mobile technology started in the late 1980s. Compared to its predecessor, this had better capacity and coverage. The capacity increased because it started using digital voice transmissions that could be multiplexed more effectively. Hence it penetrated more into the market. Moreover, the conversations transmitted are encrypted. It also introduced newer services for mobile, like roaming, SMS (Short Messaging Service) text messaging and data. GSM [7] (Time Division Multiple Access based) and CDMAOne (Code Division Multiple Access based) are well known standards associated with this technology. 2.5G (second and a half generation) and 2.75G bundled GPRS (General Packet Radio Service) and EDGE (Enhanced Data rates for GSM Evolution) respectively along with GSM for data transmission. EDGE is standardized by 3GPP [8] as part of the GSM family. These data services can be used for WAP (Wireless Application Protocol) and MMS (Multimedia Messaging Service) services as well as Internet access. 2.5G could provide data rates from 56 kbit/s up to 115 kbit/s.

7. http://www.gsma.com/aboutus/gsm-technology/gsm
8. http://www.3gpp.org

1.4.1.2 Third Generation Technology(3G)

3G technology provides added bandwidth, supports multiple mobile applications and clarity of digital signal i.e. better Quality-of-Service (QoS). Networks using this technology can offer a data transfer rate of at least 2Mb/s. There were various standards for developing the networks all over the world. The current one provided a family of standards which could provide services independent of the technology platform and whose network design standards are same globally. The typical 3G standards include W-CDMA (Wideband Code Division Multiple Access), CDMA2000, TD-SCDMA (Time Division Synchronous Code Division Multiple Access). IMT-2000 is a family of technologies for 3rd Generation mobile communications, and has been defined by the International Telecommunication Union (ITU). The objectives of this family concept include an assurance of global roaming and interoperability between the various technologies. The technologies themselves are specified by the 3GPPTM and 3GPP2 partnership projects, and by ETSI [9]. It provide application services like wide-area wireless voice telephone, mobile Internet access, mobile television, GPS (global positioning system) and video conferencing to mobile users.

1.4.1.3 Fourth Generation(4G)

The 4G system aims to address the drawbacks and problems of 3G systems [EB00, Huo99, Per00]. It provides a wide variety of newer services from high-quality voice to high-definition video to high-data-rate wireless channels with seamless hand-off across multiple wireless systems and networks. One of the terms used to describe 4G is MAGIC (Mobile multimedia, Anytime anywhere, Global mobility support, Integrated wireless solution, and Customized personal service). The system supports next generation mobile services as well as fixed wireless networks. One of the major difference with its predecessors is being completely digitalized. It performs packet based switching and converges data and voice over IP. Two standard of 4G been developed and are in use: WiMAX and LTE (Long Term Evolution). Its prospective applications are IP telephony, mobile web access, HD mobile television, video conferencing, gaming across physical locations, cloud computing, virtual navigation, tele-medicine, online education etc.

Fifth Generation(5G) is still futuristic technology. But there exists few literature where have authors have tried to make a prediction and provide some insights about its trend. “The 5G (Fifth Generation Mobile and Wireless Networks) can be a complete wireless communication without limitation, which bring us perfect real world wireless - World Wide Wireless Web (WWWW)” [Sha13].

We have described the consecutive generations of mobile communication. Apart from their services which improved due to better technology, they found increasing use in M2M communications. In the remaining part of this subsection 1.4.1, we illustrate how these generations were used for M2M communications.

Role of Mobile Technology in M2M Communication

9. http://www.etsi.org

Mobile technology provides long range wireless communication. From the evolution of its generations, with the improvement of data communication capabilities, the bandwidth increased due to deployment of newer technologies; novel services were also added. This facilitated the ubiquitous devices to exchange information, wirelessly and remotely. The SMS text messaging and packetized data in 2G technology were used by remote devices to transmit pieces of information to its peers or remote servers for data processing. FoneAstra, is a mobile phone-based temperature system deployed to monitor the national vaccine cold chain in Albania [CBA]. Different vaccine storage facilities across locations have temperature sensors installed. The phone connected to each facility communicates with the central server using SMS messages. The central database stores the temperature data received while the server has the capability to respond with command messages to the facilities e.g. alarm messages. The higher bandwidth provided in 3G and 4G networks enabled devices to exchange greater amount of information e.g. sensors sending continuous measurement data.

The current generation (4G) of mobile technology has an IP based approach. Hence the devices connected to the Internet would possess an unique identifier i.e. IP address. These devices are no longer limited to sensors. They could be any physical object with embedded systems that wants to communicate; like coffee machine, printers, medical equipments etc. This enables to address them uniquely over the network for communication by other devices. Imagine a situation that before leaving office, we can switch on the rice cooker and room heater at home, using an application on the smartphone. Alternatively, it could be the phone performing the job automatically, when the application detects itself within 3 miles of home location. The devices connected onto the network together are called Internet of Things (IoT).

Apart from the above advantages, an estimate of a device's location could be obtained using the communication technology. Localization is calculated either via multilateration or multiangulation of radio signals between (several) radio towers of the network and the phone, or simply via GPS (Global Positioning System). Physical locations could also be determined using geolocation systems if the device is connected using an IP address. However, they are not always accurate as there still exist limitations. These approaches of outdoor localization with mobile technology are used for pervasive application areas related to location based services e.g. content delivery, object tracking etc. It is different from indoor localization using short range communication (like bluetooth, infrared etc.) in terms of approach as well as application areas. It would be appropriately discussed later in this section.

1.4.2 WLAN for data communication

Wireless Local Area Network abridged as WLAN or Wi-Fi are the wireless alternative of the traditional cabled LAN. It has become a very popular technology as it enables people to gain locational mobility using portable devices like smartphones, PDAs (personal digital assistants) and laptops. WLANs provide high speed data communication in small areas such as a building or an office. It allows users to move around in a confined area while they are still connected to the network. The universal standards for WLAN was developed by the IEEE Project 802.11. The 802.11 work-group currently documents use in four distinct frequency ranges: 2.4 GHz, 3.6 GHz, 4.9 GHz, and 5 GHz bands [Ass]. Each range is divided into a multitude of channels. Countries apply their own regulations to both the allowable channels, allowed users

and maximum power levels within these frequency ranges. The minimum bandwidth needed for operations such as file transfer and program loading is 1Mbps. Operations which need real-time data transmission such as digital voice and process control, need support from time bounded services.

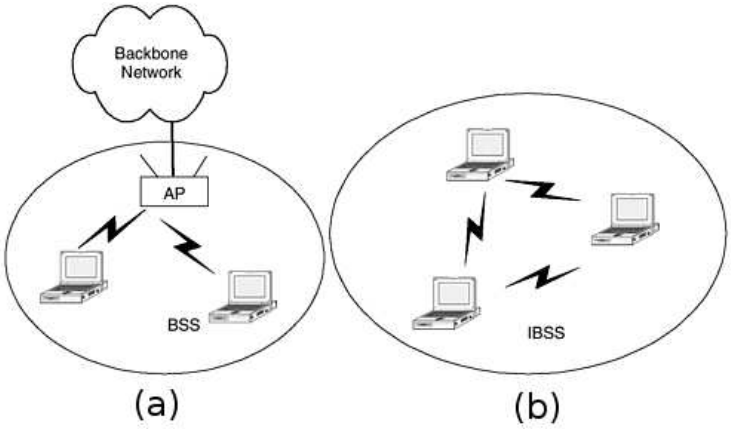

Figure 1.7: (a) Infrastructured Wireless LAN; (b) Ad-hoc Wireless LAN.

Basically there are two standard modes of configuring WLAN. Infrastructure mode is one, where the permitted devices such as phones, laptops connect to the wireless network with the help of Access Point (AP). Wireless Access Points are generally routers which are used to connect with computer via Ethernet port. SSIDs are used to configure an AP with the additional use of security key to prevent unauthorized access to the network. This mode is available more evenly, scalable, easy to supervise and has enhanced security. Peer-to-peer or Ad hoc mode is the other type of connectivity using direct communication between group of devices. It does not require any AP. Figure 1.7 illustrates the two modes, pictorially.

WLAN infrastructure mode is frequently used to provides service for internet connectivity in private homes, high street chains and independent businesses, as well as in public spaces as large as campuses have Wi-Fi hotspots set up either free-of-charge or commercially. The later is used as a limited wireless network which invites people in the same room for exchanging data.

1.4.3 Short range communication technology

Short range communication refers to wireless technologies that ranges from few centimeters to a hundred meter. In the pervasive computing context, short range communications are used for physical sensing purposes unlike the applications of long range mobile technology which are M2M communication, localization etc. Each of the short range technologies has a limited range of transmission which guarantees proximity of the communication objects. This aspect is leveraged by pervasive and ubiquitous applications in various ways e.g. context sensing, interacting objects, security etc. Two NFC (Near Field Communication) phones can

exchange business cards simply by touching them together. The communicating range of NFC is less than 30cm and so the phones require to be very close. This gives the users a feeling of exchanging information and interaction through physical contact. Additionally, it would also be secured due to the physical proximity and would be impossible for an intruder to appropriate the information.

While wireless networks like WLAN (802.11) focus on data communication among devices within the range of a building or campus or metropolitan area, there are other situations that require lesser coverage areas. Typical applications that require such small range wireless connectivity are peripherals to a mobile device, forming network of sensors or integrating the components of a home theater system. The components form a wireless personal area network (PAN) within a signal range of 1 meter to 100 meters having a wide variety of data rates. This type of communication also consumes low power given the type of energy constraint devices.

The next few sections discuss the existing technologies for low range communication. They have different standards which could be chosen depending on the application area. The first three among them (i.e. bluetooth, zigbee and infrared) are also considered as WPAN technologies.

1.4.3.1 Bluetooth

Bluetooth was derived as a short range wireless communication standard by IEEE Project 802.15.1 [Grob] based on the Bluetooth™ foundation specifications [Groa]. Its Core Specification mandates a minimum range of 10 meters or 30 feet, but there is no set limit and manufacturers can tune their implementations to provide the range needed to support the use cases for their solutions [10]. It provides low-cost wireless solution between small form factor devices at a short range. Unlike Infrared (IR), the devices need not be in the line of sight to communicate. Its radio link can provide fast and reliable transmission of both voice and data. Additionally, it can carry up to three high-quality voice channels simultaneously at speeds to 1 Mbit/sec, even in noisy environments.

Unlike WLAN, Bluetooth does not require prior setup and easy to configure quickly for sharing resources, transmission of files or creating audio links as in Bluetooth hands-free or headsets. It connects devices such as PDAs, digital cameras, mobile phones, laptops and computers, printers, fax machines, game consoles [11] together. The devices advertise for all the services they can provide which can accessed by other devices within its range without worrying much about network addresses, permissions etc. If devices are enabled in auto discovery mode, they would be able to see (and pair with) each other, if within transmission range. Then, given the users permission, they tend to form small ad hoc networks called Piconets, where two or more devices share the same channel. One of them acts as a master and the rest connected to it acts as slaves. A Scatternet is formed with multiple Piconets, with a slave acting as a gateway between two. In this case, TDM MAC is used to share the communication channel. The technology is designed to consume very low power and the radios switches off when inactive.

10. http://www.bluetooth.com/Pages/basics.aspx
11. Nintendo Wii, Sony PS3, Xbox 360 etc. uses Bluetooth controllers

1.4.3.2 ZigBee

There exist various low powered devices, that are supposed to work autonomously for months or years without recharging. For these devices, Bluetooth was still inefficient and power hungry. Hence the IEEE 802.15.4 [Groc] standard was specified. The specification describes how these small, low powered digital radios uses beaconing technique which is an advertisement of a node's presence to others around using small packets. Then it tries to form a network by connecting to the other devices and letting them joining in. ZigBee (based on the specification mentioned above) protocols use an ad-hoc, on-demand, distance vector to automatically construct a low speed ad-hoc network of nodes. In most large network instances, the network will be a cluster of clusters. It can also form a mesh or a single cluster.

There are several frequency channels allotted for ZigBee to operate. Like Bluetooth, these channels are used to avoid interference with neighboring networks. Each ZigBee network selects a particular frequency and operates using that frequency during its lifetime. Unlike Bluetooth, they don't perform frequency hopping. ZigBee operates in the industrial, scientific and medical (ISM) radio bands (868MHz in Europe, 915MHz in countries such as USA and Australia, and 2.4GHz in most jurisdictions worldwide). The technology is intended to be simpler and cheaper than other Wireless Personal Area Networks such as Bluetooth. This is achieved by pursuing communication protocols that require lower data rates and low power consumption. ZigBee's current focus is to define a general-purpose, inexpensive, self-organizing mesh network that can be used for industrial control, embedded sensing, medical data collection, smoke and intruder warning, building automation, home automation, etc. The resulting network will use very small amounts of power so individual devices might run for a year or two using the originally installed battery.

The application areas of ZigBee are different than Bluetooth because of their behavioral design. Consumer devices use Bluetooth for local communication which can wake up and get a packet across the network in about 3 seconds, when powered on. In contrast, ZigBee could perform the similar task around 15 milliseconds.

1.4.3.3 Communication using the light spectrum

Infrared

Infrared (IR) is another mode of data communication used for portable devices like mobile phones, PDAs located within short range. The devices are usually manufactured according to the standard specified by IrDA [IrD]. The devices using Infrared like remote controls which are very common, have infrared light emitting diodes (LEDs) to emit infrared radiation which is focused into a narrow beam using a plastic lens. The data is transmitted as a modulated beam. The receiver converts the received infrared radiation to an electric current using a silicon photodiode. It responds only to the rapidly pulsing signal created by the transmitter, and filters out slowly changing infrared radiation from ambient light. Infrared communication is more useful for indoor use. Unlike the other short range communication technologies, Infrared works in line of sight between the sender and the receiver. It does not penetrate walls, thereby being non-interfering with devices in the neighboring room. This differentiates from its peer technologies

for object interactions in pervasive applications.

A very commonly used ambient application of Infrared is operating the opening and closing of doors in buildings, elevators for objects and human presence. It utilizes the property that Infrared works only in line of sight. Fig 1.8 shows the working how a screen of infra-red beams acts as an invisible safety curtain across the door entrance [12]. When an obstacle breaks this screen, the system detects it and immediately makes the doors retract.

Figure 1.8: Automatic doors using Infrared

The Infrared technology is highly directed in nature. This can be used for addressing individual objects. It gives a perception of interaction on a one-to-one basis. A classical example is a remote control, when used to operate various electronic appliances temporally and spatially. Furthermore, beams of infrared can be used to form an interaction space for sensing. Figure 1.9 presents the layout of a room where it shows two such spaces named “Infrared A” and “Infrared B”. Such disjoint spaces can be engineered within the same room, using several infrared beams and non-interfering with each other.

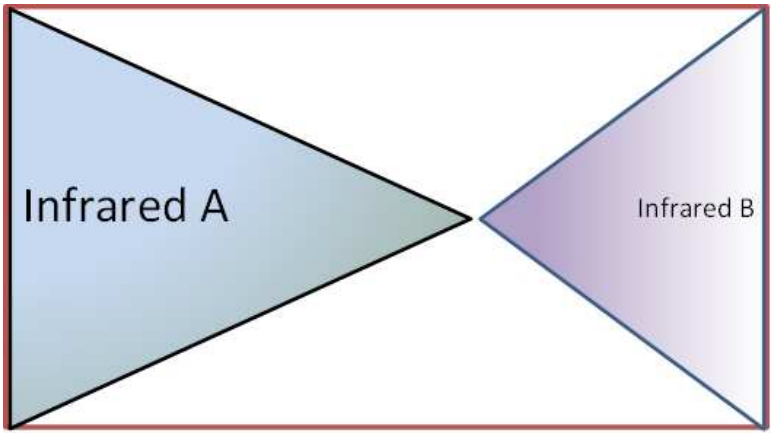

Figure 1.9: Interaction spaces using infrared beams

12. Picture shows Lambda door detector designed by Otis. http://www.otis.com/site/kw/pages/Mod_DoorDet.aspx

Visible light communication

While infrared technologies have been used for communication for a long time, usage of visible light waves for transmitting high volumes of data over short distances has been found attractive recently. The data is encoded by modulating the amplitude of light. The modulation is done thousand times faster than the perception of human eye to be detected. It can serve the dual purpose of illumination as well as communications. Experiments have achieved to demonstrate data rates to the order of Gbit/sec. Duplex communication is convenient with point-to-point links. However usage of illumination sources suits naturally to data broadcasting. Its current challenge is providing an up-link to the transmitter or the light source.

1.4.3.4 Ultra Wideband (UWB)

UWB is a technology used for transmitting information over short distances at high data rates exceeding 100 mbits/s, spread over a large bandwidth ($>$ 500 MHz) at a low power range. It aims to provide users with shared spectrum for transmitting data. Federal Communications Commission (FCC) in the United States has authorized the unlicensed use of UWB in the 3.1 to 10.6 GHz frequency range. This is intended to provide an efficient use of scarce radio bandwidth while enabling both high data rate wireless connectivity within Body Area Networks (BANs), PANs and within buildings and at longer range, low data rate applications, as well as radar, collision obstacle avoidance, precision altimetry and imaging systems [Kal07]. While the conventional wireless systems that need to use baseband signals to modulate radio frequency (RF) carrier signals, UWB can be used to directly transmit signals at baseband frequencies. In pulse based UWB, the transmitter only needs to operate during the pulse transmission, producing a strong duty cycle on the radio and minimizing baseline power consumption. As most of the complexity of UWB communication is in the receiver, simple, low power transmitters can be supported.

UWB characteristics are well-suited to short-distance applications, such as PC peripherals. Due to low emission levels permitted by regulatory agencies, UWB systems tend to be short-range indoor applications. Due to the short duration of UWB pulses, it is easier to engineer high data rates. Suitable applications exploiting this characteristic could be chunks of file transfer. UWBs precision with real-time location, low power and short period is ideal for radio sensitive places like hospitals. Application like "see-through-the-wall" precision radar-imaging has been proposed using UWB [13] [14].

1.4.3.5 RFID

Radio-frequency identification (RFID) is a wireless data capturing technique used for automatic identification of objects. It uses RF waves for data communication. Identifying objects are necessary as the first step, to decide on collecting data about them and/or using the data for wide range of applications in computer systems. The applications vary from process automation to security control. Kärkkäinen and Ala-Risku has identified four basic classes of

13. http://www.timedomain.com/news/wall.php
14. http://ukgrads.thalesgroup.com/Files/TRT UWB radar.pdf

applications: Authentication, tracking, process effectiveness, and information management applications [KAR03].

- Authentication applications are used for verifying the identity of objects (animate or inanimate).
- Objects are very frequently required to be located along with their route traversed and the current state. These item tracking and/or traceability applications are common for business process, logistics and defense forces. They are mostly performed online being connected to a central database.
- In process effectiveness solutions, automatic identification is used to automate processes or to automate the data input to save time and reduce errors. The most common applications of automatic identification have been in industrial applications, especially in warehousing and manufacturing, and in retail applications, such as sales-point scanning and inventory management.
- In information management applications, identifiers are used to access information concerning the objects the identifiers are attached to. The information concerning the items can be stored either in the identifier or in a database. When the information is stored in a database, the identifier is used as a reference to access the information. There are several levels on which the item information can be managed. The most generic level is the product functional level, which is used for design and usage purposes. In logistics applications the common alternatives for information management are at the product type level, production batch level, and product individual level.

Depending on the application class and requirement, bar codes (linear one-dimensional, two-dimensional and composite), Radio Frequency Identification (RFID), biometrics, contact memory technology, magnetic stripes, Optical Character Recognition (OCR), smart cards, and voice recognition are the typically used technologies. The authors in [KARK01] provides detailed description of these technologies. There are multiple parameters that determine the choice of technology for the user's specific application; some of them like data capacity, ease of readability (under conditions), simultaneous read, cost (installation, operation and maintenance) etc. Fig 1.2 shows a comparative overview of the abilities among various technologies while being used for the application classes. It substantiates the fact that for automatic identification technology bar coding is an established one, RFID is its upcoming tough competitor. It is already gaining popularity - in pervasive computing - by physical objects being RFID tagged as a means of connecting them to the virtual world. Preradovic and Karmakar mention strongly that chipless RFIDs (in research and development) would be "Bar Code of the future"; since the cost of its embedded chip is limiting the growth [PK10]. This thesis is motivated by these facts along with a partial futuristic vision that RFIDs would attain greater momentum and would be "The Technology" in its league. Most of the prototypes proposed later use RFIDs. So, for the convenience of the reader the next few paragraphs would describe RFID in detail.

Typically, a RFID system consists of three major components: an interrogator or reader, a RFID tag or transponder and middleware software. The reader sends interrogation signals to RFID tags for identifying; the tags containing an identifying code and some data. The middleware software maintains the interface between the reader and the computer; and different protocols to encode or decode the information exchanged between the tag and the reader. The computer attached to the reader is very often networked to remote database for further process-

Application⟹ Technology⇓	**Authentication**	**Tracking**	**Automation**	**Information Management**
Barcode		X	X	X
RFID	X	X	X	X
OCR			X	
Biometrics	X			
Vision Recognition			X	
Magnetic Ink	X		(X)	
Smart cards	X			X
Contact memory			(X)	X
Magnetic stripe	X			X
Bluetooth	(X)	(X)	(X)	(X)
GPS		X		
GSM cell location		X		

Table 1.2: Different identification technologies classified by application types [KAR03]

ing. A RFID reader has a range for querying tags. Fig 1.10 presents the block diagram of a typical RFID system.

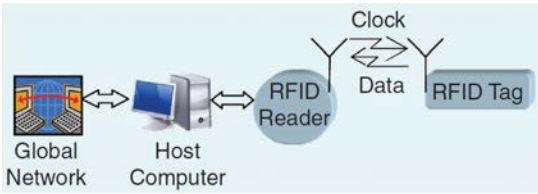

Figure 1.10: Block diagram of a typical RFID system [PK10]

RFID tags

RFID tags are attached to objects for being able to be identified automatically. Tags come in various shapes, sizes and formats; the appropriate one needs to be selected to suit the application purpose. Most of the widely used ones are made up of the following components: an antenna (made from conductive material), a small silicon chip for control and memory and a power system. Fig 1.11(a) shows the various form factors that are used depending on the application whereas fig 1.11(b) exhibits two RFID tags of different sizes along with their similar components inside. Tags are categorized based on their type of component used as explained in Table 1.3.

Since active and semi-passive tags are costlier than passive tags, they are used for tracking high-value goods or higher stakes. Other parameters such as durability, operating frequency,

Type of Tag	**Characteristics**
Classification by presence of Chip	
Chip	- Cost of tag incurred mainly due to the chip; its design, manufacturing and testing. - RFID tag (US$0.10) costs still higher than optical bar code. - In spite of cost, its widely used for applications.
Chipless	- RFID tags without the use of traditional silicon ASICs to lower cost. - Research in progress. Mostly reported as prototypes and very few are commercially viable or available. - Expected to cost below US$0.01. - Most chipless RFID systems use the electromagnetic (EM) properties of materials and/or design various conductor layouts/shapes to achieve particular EM properties/behavior.
Classification by Power Source	
Passive	- The incoming RF signal from the reader turns power system on by inducing a current in the tag's antenna. - Usually smaller in size, mechanically flexible, and much cheaper compared to other types. - Has a longer shelf life and is currently the most commonly used. - Tags can be of any frequency bands.
Semi-Passive	- Contains a battery and requires the reader's power to transmit message back to the RFID reader. - Has a longer shelf life than active tags and are usually used to track valuable assets. - Tags are usually of the UHF frequency band.
Active	- Contains a battery and sends signals automatically to the RFID reader. - The most reliable and offers the greatest communication distance. - The most expensive but are typically used in logistic applications. - Tags can be of UHF or Microwave frequency bands.
Classification by Memory Type	
Read Only	- Information on the tag is factory programmed, and the memory is disabled to prevent any future changes. - Only a limited amount of memory can be stored (96 bits).
Read-Write	- Information can be read and flexibly altered by the user. - Typically contains more memory (32k Bytes to 12k Bytes) and more expensive than read only tags.

Table 1.3: Classification of RFID tags [IEI, CNR, PK10, Fle02, Wan06]

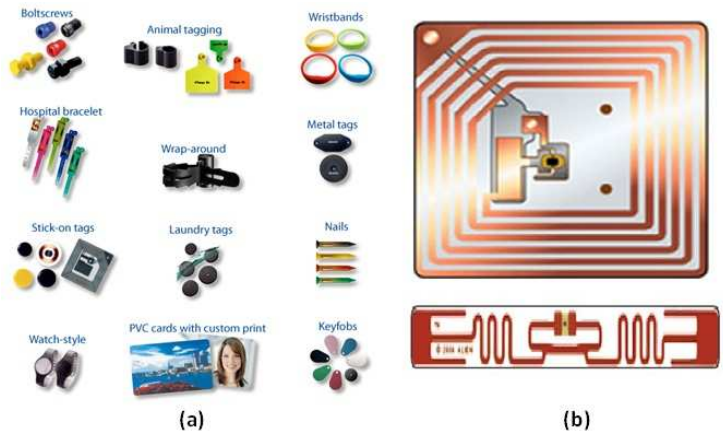

Figure 1.11: (a) RFID tag form factors (b) Inside RFID tag

re-write ability, memory size and regional regulations are considered while selecting a tag for application purpose.

RFID Interrogator or reader

It's basically a radio frequency (RF) transmitter and receiver which is controlled by a microprocessor or digital signal processor. It communicates with the RFID tags or transponders to capture their stored data; could be identifier or other information. The reader - which has an attached antenna - sends radio energy to the tag and waits for the tag's response. In simple systems, the reader's energy works as a trigger to activate tags for sending response with data. For others, the reader sends commands to the tags for operations such as memory read, write etc.

Table 1.4 describes the classifications based on their design and method of fixation.

Type of Reader	**Characteristics**
Classification by Design	
Read	- Can only read information from RFID tags.
Read / Write	- Can read and write information from and on RFID tags.
Classification by Method of Fixation	
Fixed	- The device is fixed or attached at a fixed point; e.g. at the entrance gate.
Mobile	- The device is mobile and can be carried and operated by the user regardless of location

Table 1.4: Classification of RFID Readers [IEI]

Reader antennas are often the most complex to design because principles adopted by different frequency ranges (LF, HF, UHF) are fundamentally different. Moreover, most antennas need to be tuned to the resonance of the operating frequency, which makes them vulnerable to external effects and can greatly impact communication distance.

Frequencies and Protocols

RFID is centralized on wireless communication and makes use of radio waves that are a part of the electromagnetic spectrum. RFID operates in the unlicensed spectrum space (also known as ISM: Industrial, Scientific, and Medical); however, exact frequencies may differ depending on regulations in different countries. Nevertheless, these operating frequencies are generally categorized into four frequency bands globally. Table 1.5 summarizes the characteristics of each band with their respective applications.

Band	**LF (Low Frequency)**	**HF (High Frequency)**	**UHF (Ultra High Frequency)**	**Microwave**
Frequency	30 - 300 kHz	3 - 30 MHz	300 MHz - 3 GHz	2 - 30 GHz
Typical RFID Frequency	125 - 134kHz	13.56 MHz	433 MHz, 860 - 960 MHz	2.45 GHz~
Communication Range	< 0.5m	$\leq$ 1.5m	433MHz: $\leq$ 100m 860 - 960MHz: 0.5m~5m	$\leq$ 10m
Data Transfer Rate	< 1kbit/s	25kbit/s	30kbit/s	$\leq$ 100kbit/s
Characteristics	Short range Low data rate Penetrates water but not metal	Mid range Reasonable data rate Penetrates water but not metal	Long range High data rate Cannot penetrate water or metal	Long range High data rate Cannot penetrate water or metal
Application	Animal ID, Access Control, Car Immobilizer	Smart Labels, Contactless Travel Cards, Access Security, Apparel	Logistics, Baggage Handling, Electronic Toll Collection	Electronic Toll Collection

Table 1.5: RFID Frequency Characteristics

In addition to a selected frequency band, the RFID tag and reader must both comply with a specific protocol for communication to occur between them. To put it simply, these protocols define the language used between the tag and reader; thus, both components should support the same protocol in order for the two to understand and communicate with each other. There are numerous protocols established by ISO and EPC (Electronic Product Code) Global, however, the most commonly found protocols are listed in the Table 1.6 below.

RFID tags and readers work in unison and correspond part of the same system. Various applications domains like transportation, logistics, manufacturing, processing and security use RFID systems, often with additional infrastructure like communication, networked computers, database etc. for providing decision making and business planning. Table 1.7 displays a typical

LF (125 kHz)	**HF (13.56 MHz)**	**UHF (860 - 960 MHz)**	**Microwave (2.45 GHz)**
ISO11784/5, 14223, ISO18000-2	ISO14443, ISO15693, ISO18000-3	ISO18000-6, EPC Class 0/1, EPC GEN II	ISO18000-4

Table 1.6: RFID Protocols/Standards

list of RFID use cases.

Animal tagging	Road toll collection and rail application
Animal husbandry	EAS (Electronic Article Surveillance)
Toxic and medical waste management	Access control
Postal tracking	Time and attendance
Airline baggage management	Manufacturing processes with robotics
Paper money anti-counterfeiting	Monitoring of offenders
Anti-counterfeiting in the drug industry	Passports
Vehicle immobilisers and alarms	

Table 1.7: Typical applications of RFID [Rob06]

EPC specifications have defined five classes of tags based on the functionality, as shown in table 1.8.

Apart from the traditional uses discussed above, RFID is increasingly used for ubiquitous and pervasive applications. It is due to its various properties that are utilized innovatively. The first among them is its ability to address tagged objects individually. Undoubtedly, these objects should be within the range of the interrogator. Multiple readers could be combined for “spatial” coverage of an area. Figure 1.12 demonstrates the idea of such a zone between two RFID readers. Every tagged objects passing through this area can be interrogated and interacted. The area is so precisely covered that objects are not detected, when on the periphery. The manner is partially similar to Infrared interaction space discussed earlier. However, RFID has advantages compared to Infrared. A RFID reader can choose to address single or multiple objects simultaneously. Batch processing and inventorying are the traditional applications that handle multiple tags.

RFID tags provide an identifier to each and every physical object. This enables pervasive applications to single them out for interaction. A tagged object is automatically detected when brought within an interrogating area. It’s unlike most short range communication like bluetooth, Wlan etc where user’s active participation is required even when within their transmission range. The tags can respond autonomously. This behavior is exploited as actuation within a “spatially” covered area for pervasive applications. Lastly, the tags contain memory that provide storage space for pieces of information like object description to be read by an interrogator. All these features of RFID help pervasive applications to sense the physical world situation.

Class	Nickname	Memory	Power	Features
0	Anti-Shoplift Tags	None	Passive	Article Surveillance
1	EPC	Read-Only	Any	Identification Only
2	EPC	Read-Write	Any	Data Logging
3	Sensor Tags	Read-Write	Semi-Passive or Active	Environmental Sensors
4	Smart Dust	Read-Write	Active	Ad Hoc Networking

Table 1.8: Tag classes. [Wei03]

Figure 1.12: Using two RFID readers for “spatial” coverage

Until now in this long section 1.4, we have discussed the state of the art for wireless communication technologies. We would conclude the section with a summarizing discussion on the fundamental properties that are desirable for building pervasive computing systems.

Analysis: Wireless communication in the context of pervasive computing

Wireless communication has pervaded into our everyday lives. The reducing cost of equipments, smartphones and easy availability of applications has allowed the easy penetration of communication technologies. However for pervasive computing purposes, long range technologies play the role of transmitting information between machine-to-machine (M2M). The case is different for short range transmissions. Apart from transmitting signals and communicating data, these technologies enable ubiquitous sensing and actuation; such as presence detection, identity recognition, and more. These technologies are used individually or in combination for applications, depending on the requirement. The star symbols in table 1.9 shows how the transition of the usage takes place in pervasive applications context. It shows their contrasting usage between machine-to-machine (M2M) communications and pervasive application enabler. Long range technologies are mostly used for M2M with high data rates over long distances. As we move on towards lower operating range and bandwidth, they are lesser used for data communication and more favored for proximity services. IR and RFID are popularly used for physical interactions due to their lowest operating range.

	Cellular (4G, 3G...)	**WLAN (WiFi)**	**WPAN (Bluetooth, Zigbee, IR)**	**IR, RFID**
Data Communication (M2M)	⋆⋆⋆⋆	⋆⋆⋆	⋆⋆	⋆
Bandwidth	Very high (in Mbit/s)	Very high (in Mbit/s)	Medium (~1 Mbit/s)	Low (< 115 Kbps)
Operating Distance	from tens to thousands of KMs	< 500m	< 100m	< 10m
Pervasive Application	⋆ Geolocation	⋆⋆ Proximity Service	⋆⋆⋆ Proximity Service	⋆⋆⋆⋆ Sensing/Actuation

Table 1.9: Transition of usage w.r.t. Pervasive Applications

Figure C.2 presents the transition of usage pictorially. The communication range increases as we move up diagonally from bottom-left of the picture. The objects within the smart home uses short range technologies for sensing and interacting. WLAN and WiMAX are have intermediate usage between approximate sensing of physical environment and communication withing their covered range; like in and around the house. The long range technologies are completely dedicated to talk to other objects located remotely outside the house.

Pervasive applications leverage the properties of short range communication technologies. Hence, it is important to know them and use appropriately for sensing and interacting with physical objects. Table C.1 summarizes them. Identification helps to recognize the objects, which is essential while the addressing mode determines the number of interactions that could be handled simultaneously within the coverage space. The propagation property of communication technology is used to engineer the spatial coverage and style of interaction. Recall that

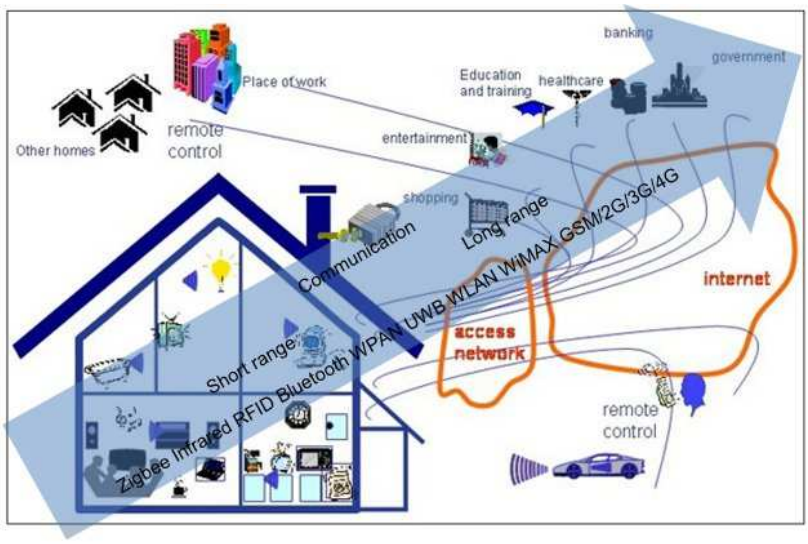

Figure 1.13: Long and short range communication

infrared communicates in line-of-sight, which makes object interactions directed. For other technologies, it is enough for the interacting objects to be within their communicating range specified in the last row of the table.

	WPAN	Bluetooth	Zigbee	Infrared	UWB	RFID
Identification	X	X	X	X	X	X
Addressing mode	Multiple	Multiple	Multiple	Individual	Both	Both
Propagation	Non-line-of-sight (NLOS)	NLOS	NLOS	Line-of-sight	NLOS	NLOS
Range	1m-100m	upto 10m	10m-100m	upto 10m	upto 30m	upto 10m

Table 1.10: Properties: Short range technologies

There are few related works that illustrate these aspects [May, KhPV+13, BdS08, NSS13]. For example, small, attachable RF identification (RFID) tags could help to determine the approximate position and identity of the tagged object or individual in a space equipped with specialized readers [WHFG92]. The "Tile"™ [15] is a commercial product that makes use of this concept.

15. Tile, the world's largest lost and found. http://www.thetileapp.com/

1.5 Information architecture in Internet of Things (IoT)

1.5.1 Tags data structure

The inception of idea to explore RFID could be traced back around the 1940s from the paper "Communication by means of reflected power", by Harry Stockman [Sto48]. The radar and radio research undertaken during the Second World War, led to the exploration of this new area. Thereafter, lot of developments were performed until they were first begun to be used for commercial activities in the 1960s. Sensormatic, Checkpoint and Knogo were the companies founded in that decade which developed electronic article surveillance (EAS) equipment to counter the theft of merchandise [Lan05]. Typically, these tags are attached to items in retail stores and an alarm is triggered if goods are taken out before their attached tag is deactivated. The EAS systems are considered to be the first widespread use of RFID. They often use one bit memory tags for detection of their presence or absence and their inexpensive production provided cost-effective security measures. Tags with multiple bits were still being researched. They were uncommon due to their large size compared to single bit tags.

In the next three decades (1970s to 1990s), RFID saw a great deal of research and development from individuals and organizations all over the world. Advancements were made and they started finding use for identification and tracking. Such RFID uses started to be realized with livestock tracking and later extended to various application domains. The tags were mostly passive with an operational range of tens of meters. For unique identification of objects, a naming system has been developed. It is called electronic product code (EPC) and is intended to be universally accepted as a means of link between the physical objects and computer network. It serves as a unique name for reference in information system and is "burned" permanently into the memory of RFID tag as fixed binary number. Presently, EPCs are either 64 or 96 bits long unique identifiers that encodes the manufacturer, product type and the product serial number of physical objects. Figure 1.14 shows the standard format of a 96 bit identifier. EPC of an item by virtue of the RFID tag affixed to it does not change throughout its lifetime.

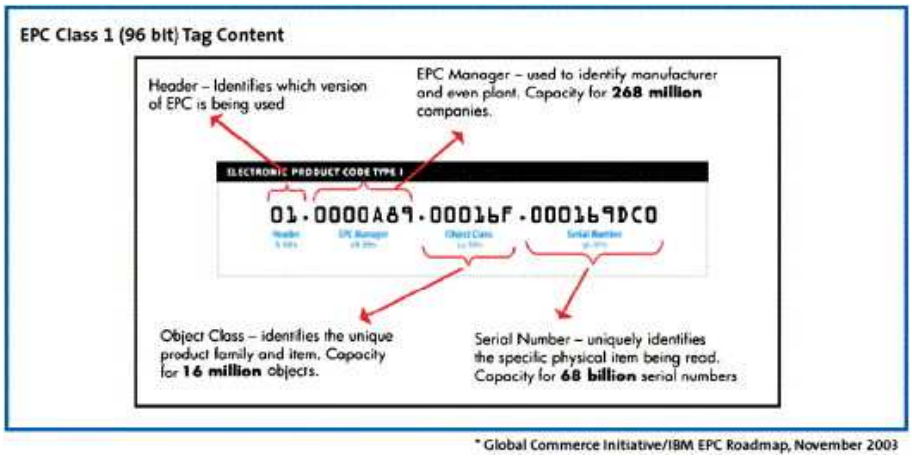

Figure 1.14: EPC Tag Format [Rob06]

EPC identifiers had been designed to pinpoint every physical object uniquely when compared to barcodes that typically identify the manufacturer and class of products. For example, every soda bottle manufactured can be addressed individually using EPC whereas they will have identical barcodes printed on the labels. The tags are typically designed to reply with the fixed length EPC when queried by a RF reader. This EPC information is passed on to various processing devices in the backend infrastructure for decoding and take relevant actions. Currently, there are components as savant, object name service (ONS) and physical markup language (PML) server decoding the tag [CTCC03]. ONS serves as a link between the EPC and the data files containing information related to the tagged item. The ONS is an automated service that returns network addresses of one or more data files when provided with an EPC. ONS has capabilities to parse EPC to retrieve its specific information fields to fetch the appropriate data locations. Its concept is similar to DNS (Domain Name System) that identifies website addresses corresponding to website names. Finally, the PML is an XML-based language that acts as a database of information about physical objects. It contains object descriptions that are both human and machine readable.

Recently, RFIDs are being used for identifying and linking physical items to digital object memories (DOMe). A DOMe is a digital storage space intended to keep permanently all related information about a real physical instance - e.g. its use, history or previous object-related events - that are collected during its lifespan. This concept is demonstrated for different application domains [SMK+10, SK08, KGB+11].

RFID tags has a clear advantage over barcodes due to its identification capabilities using radio communication and without requiring a line of sight between the object tag and the interrogator. Hence, objects packed in boxes or hidden could be identified without being revealed. Apart from such identification of objects, its tag memory is used for storing additional information. Some simple examples of simple static information stored on tag memory are manufacture/expiry dates of objects, basic information while animal tagging etc [VPS+10]. Dynamic information concerning traceability and monitoring of physical objects are sometimes stored into tag memory. It is very commonly used for perishable items like food [APN+09]. This approach is analogical to the scheme of digital object memory stored entirely and locally on item tags [BKH13]. However the limited nature of tag memory poses a challenge of storing the volume of information. It can be tackled by adopting solutions like centralized archival, forgetting, abstracting and compressing [FKK09].

1.5.2 Data processing architectures

Presently, most of the architectures using tagging solutions are centralized. The physical situation is communicated to the ubiquitous computers for processing. Interactions with physical objects are identified using different kinds of unique marker systems such as RFID, barcodes, QR codes, URI's etc that requires look-up in information systems before further activities. The smart box is one of the earliest concepts that monitor the contents inside a container using RFID for identification [FLS03]. The market demand for absorbing these identification technologies for efficient supply chain management has prompted various IT vendors to adopt accordingly. While Microsoft's Axapta, Great Plains and Navision systems are designed to be RFID-ready softwares, SAP and Oracle have provisions for RFID [Rob06]. As

discussed in the previous subsection, digital object memories (DOMe) are a recent trend that are digital representations of physical objects resident in central information systems and take part in interactions. Internet of Things (IoT) - mainly central processing based architecture - which is being used extensively utilizes DOMe for representing physical entities in the cloud [Hec10, BdJHS12].

There are very few work in the literature that perform decentralized data processing. RFID tags are used as local storage spaces for information linked to the tagged physical object. This setup allows autonomous processing of data for every individual or a group of item(s). Consider for example perishable items tagged in a cold chain that stores log of the environment factors (temperature, humidity etc.) at every stage. In this way, it could be verified from the data for any irregularities anytime autonomously. The application is demonstrated in [YL09]. Integrity of objects is another physical property that has been demonstrated autonomously using RFID tags [CBA11]. The memory space of objects tagged are used to link between themselves locally to preserve integrity. This concept is demonstrated in Ubi-Post, that ensures the delivery of high value goods [VBC12].

Among both the data processing architectures described above, the first is prevalent and the later very rare. The authors in [DMS07] have called them as data-on-network and data-on-tag. They have compared both approaches stating pros and cons and finally have suggested an integrated or hybrid concept with examples for better availability of data for processing. Figure C.3 tries to bring out the difference pictorially. While the continuous arrows represents a required data access, the dashed ones suggest an optional access. Subfigure C.3(a) shows the RFID tags that always require lookup on the remote database. Subfigures C.3(b) and C.3(c) represent two variants of autonomous model, where the tag memory stores data with optional data access to the remote database. While subfigure (b) represents the tags storing individual data, the other contain data linked among themselves which acts like a local storage space.

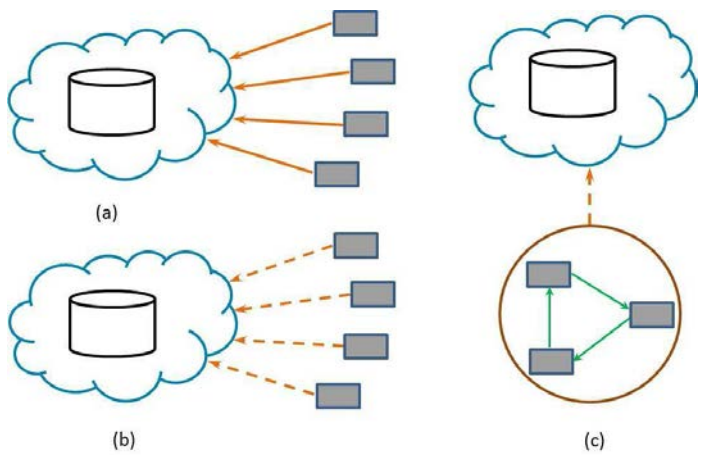

Figure 1.15: Data processing architectures

1.6 Conclusion

Pervasive computing entails embedding of processing power in everyday objects and the environment so that they can communicate information. This requires deployment of sensing and communication infrastructures enabling them to be smarter. The setup would be more effective if every physical entity is an “e-object”. Currently, the big and expensive items have such facilities to be qualified as an “e-object”. Such examples are smartphones that can participate in pervasive activities. However, the scope of pervasive computing could be widened, if the infrastructure could be extended to every object in the physical world; specially the small, inexpensive and dumb objects. Presently, incorporating these insignificant objects have not been explored much; according to our literature review. However accounting them into consideration would enable a better representation of the real world situation. We have proposed an approach to bridge this gap using RFID tags. The tags are very suitable as an inexpensive retrofit, providing an object’s presence to be sensed before actuation by pervasive systems. Additionally, this thesis has contributed with ways that utilize the tag memory to store information concerning the physical objects. The approach is a deviation from the way in which RFID tags were originally designed to work i.e. containing a link to remote information.

Chapter 2

Smart Objects with Tangible Data

2.1 Smart Environments

Ubiquitous computing research generally focuses its area of application in smart homes, smart offices or smart campus where the physical boundaries are limited to a building or a small area. The deployment of urban wireless infrastructures have enabled envisioning the development of smart cities all over the world. Building smart cities and its research involves participation from a diverse group of people which includes governments, urban planners, sociologists, and of course computer science researchers. One of the ways to realize smart cities is by augmenting existing city infrastructure with embedded sensing as well as communication and interaction technologies.

Every city all over the world is continuously in a state of flux and exhibit dynamic changes. And along with the growth of the city, their planners devise "complex systems to deal with food supplies on an international scale, water supplies over long distances and local waste disposal, urban traffic management systems and so on; (...) and the quality of all such urban inputs defines the quality of life of urban dwellers" [Sci]. Their complexities are growing increasingly larger with an ever increasing speed. These cities are made smarter to tackle these complexities. However, there is no single definition of a smart city and the meaning of "smart" depends on the context of its usage. There are many working definitions of "Smart City" from different perspectives. According to Hermenegildo Seisdedos, such cities are efficient [Sei12]. The efficiency is based on the intelligent management and integrated ICTs (Information and Communication Technologies), and active citizen participation. Caragliu et al. defines a city as 'smart' when investments in human and social capital and traditional (transport) and modern (ICT) communication infrastructure fuel sustainable economic development and a high quality of life, with a wise management of natural resources, through participatory action and engagement [CDBN09]. The authors in [NP11] presents a list of working definitions that highlight different aspects such as technology, architecture, society etc. In the context of this thesis, we choose the definition proposed by Washburn et. al. [WS09]:

"The use of Smart Computing technologies to make the critical infrastructure components and services of a city - which include city administration, education, healthcare, public safety,

real estate, transportation, and utilities - more intelligent, interconnected, and efficient."

It puts an emphasis explicitly on the use of smart computing technologies. The authors view the current urban crises as an imperative of a smart city initiative. Scarcity of resources, inadequate and poor infrastructure, energy shortages and price instability, global environmental concerns, and human health concern are some of the deteriorating conditions of cities in a crisis.

Today, efficient management and handling of waste are one among the many issues of great importance. It is required for an ecologically sustainable development in most countries. The European Union (EU) reports handling around 3 billion tonnes of waste (and still counting) each year among which, around 90 million tonnes are hazardous [Com]. So managing the waste i.e. treating and disposing them efficiently causing least possible harm to the environment is becoming difficult. It was in this context, the "Bin That Thinks" [1] project was proposed aiming for finding innovative pervasive computing solutions addressing to waste management and recycling challenges. It was financed by ANR (l'Agence Nationale pour la Recherche) for a period of three years to be coordinated among multiple teams with different expertise.

The pervasive solutions facilitate better sorting of waste for efficient recycling and reuse of their materials having economic value. The earlier the sorting, the better it is in terms of reduced contamination and overhead costs. The project proposes smart bins capable of working autonomously to deal with smart waste products. We have assumed an increase in the use of RFID tags by manufacturers/retailers for products. Extending the RFID use with the self-describing approach contributed in this thesis would make everyday used physical objects smarter; enabling them to participate in various interactions - autonomously - during its lifecycle until its disposal. Chapter 3 describes our pervasive approach to smart waste management in finer details.

This thesis - funded by the "Bin That Thinks" project - contributes with techniques to use RFIDs as small storage spaces providing information pervasively distributed in the physical space for autonomous processing. Having autonomy is one of the desirable properties but there are various challenges for achieving them. The limited memory capacity of the tags is one of them. The remaining part of this chapter elucidates our contributions considering physical objects around us, in general. The first explains "self-describing objects"; their embedded tags containing semantic description of themselves and enabling them to participate individually in pervasive applications. However, a tag memory is sometimes not sufficient to store all the necessary information. For such cases, our second contribution illustrates an approach to distribute the information over multiple tags of related objects. This forms a closed group of objects with related information and the situation is leveraged to provide a different genre of pervasive service. They are demonstrated with examples.

2.2 Self-Describing Objects

In this section, we pitch for self-describing physical objects that would enhance their autonomy. It is an unusual and novel approach from the traditional trend with RFID tags which generally store links to remote information. Currently, retailers and manufacturers utilize RFID

1. http://binthatthink.inria.fr/

tagging for inventory management. We can assume that in the future they will tag every product. Then, the tag memory could be utilized to contain description of the objects. This way of using the tags would contribute intelligence to the objects. Our proposed approach is a step forward, to put some extra bits of information that could help in taking basic spontaneous decisions related to the tagged object. They cost from a few cents to dollar [CC07, wik]. The semantic description would enable everyday used objects to participate in local interactions. However, our approach could also take advantage of the tags - used traditionally - acting as a link enabler to connect the physical objects to the digital environment.

This section has three major subsections. The first states the strong points before detailing and analyzing our approach - the Tangible Data - in the second subsection. Finally in the third, we have illustrated two examples as its proof of concept.

2.2.1 Advantages

The design of our approach implicitly benefits with the following properties.

2.2.1.1 Availability in absence network infrastructure and scalability

The trend of pervasive computing is more and more transforming the everyday used objects, smarter, with embedded technology and communication. Of course, these approaches improve their smartness than before. However, it becomes too dependent on the backbone infrastructure to exhibit the smartness. Additionally as mentioned earlier, they reduce their autonomy for independent functioning. They have a centralized architecture for storing the information which is retrieved by the objects for delivery of services.

Let's consider a coffee maker as a simple example of smart object. The smart coffee machine can detect a jar and its volume when brought for refill. It also displays the flavor of coffee that was refilled, the last time. Going by the conventional approach, the specifics of the jar and the refill i.e. flavor of the coffee, the amount and time would be transmitted and stored somewhere over the internet, each time the machine refills a jar with coffee. Additionally, the machine also requires to obtain all the information about the jar, stored remotely. We propose a modification with our alternative approach for the same application. The jars are embedded with tags storing its volume by the manufacturer. All the information of refilling a jar is written into its tag. This information can be used for displaying or refilling by any machine. Following this approach does not require the setup and use of network infrastructure. There may be situations where accessing the Internet is difficult or impossible due to location or security reasons; thus increasing the availability of the system. The cost-effectiveness is another factor that should be accounted. Embedding tags that are worth few cents can perform the same task as maintaining a storage space on the network.

Our approach can also support scalability. The increasing number of objects like coffee jars does not affect the working efficiency of the system. Since the information are attached locally to the objects, it takes the same amount of time to read their information. To provide an idea of its scale, the smart coffee maker would be highly scalable to be installed for a large social gathering without the need for any network infrastructure. Additionally, all the participants can

Figure 2.1: Smart Coffee Maker with and without using Network

have their personalized coffee mugs with embedded tags instead of jars. Apart from the these advantages, the tag would provide the added feature of being sensed by the machine to actuate (like display), when an object (container) is put in the refilling area.

2.2.1.2 Interoperability

Smart objects are envisioned to interact within their smart environments. However, it is very difficult to make them work seamlessly across all environments. The reason for it being supporting different hardware, software, security policies etc. If smart objects and their interactions are designed to function using a centralized architecture, all their services cannot be guaranteed due to restriction of access to the proprietary data over the network. One of the solutions to this problem is to duplicate and distribute the required data at various locations. However, its related challenge is to keep the copies consistent at all places. Our approach provides an alternative solution to interoperability, since the required information is coupled to the physical object. The presence of an object would guarantee its related information, stored in its tag, such that it can participate in interactions locally.

We have a classical example of e-commerce logistics where parcels pass through many hands between the sender (a business organization) and the receiver. While the parcel should be intact with its contents, it's difficult to verify them at every stage until it reaches recipient; since the sender would not want its proprietary database to be given access to the logistic companies. So our approach addresses the problem where the physical items inside the parcel are tagged by the sender. Each of these tags are stored with information about the parcel and the receiver. Logistics can use these pieces of information to check the integrity of parcel at every stage as well as verify the delivery of the complete package to the correct recipient.

2.2.1.3 Privacy and security

In the recent times, respect of privacy has become a matter of growing concern. It involves regulating information about oneself from being let out to and used by third parties for their vested interests. When centralized architectures are used, all the information resides in a remote central location, such as the cloud. The information could be variety of data captured from users like profile, behavior, location etc. These data are stored in centralized servers for large number of users, sometimes called big data. This data is manipulated, analyzed, processed or internally used for providing services. The destiny of this personal information is in the hands of third parties, once collected from the users. It is not always known on how their information could be used.

In the context of pervasive computing, using smart objects with centralized infrastructure enables to track and collect information about them. Internet of Things (IoT) promotes trend of global infrastructure where devices, sensors, readers, applications etc. have the potential of collecting various kinds of data continuously with changing environments. Sometimes, identification of objects could also lead to automatic identification of its related users [GKdPC10]. The information collected based on object identifiers, sensor data and/or networking capabilities of IoT systems might reveal information on individuals, their habits, interests, location, personal information and other preferences stored somewhere on the cloud. There are hundreds of such devices used in everyday life that collect data to provide health care, domestic functions, entertainment and many more. In some cases, the users are unaware of the fact. Using smartphone is the most common example that can cause breach of privacy. The phones have various sensors to gather data and the installed applications have access to its sensitive user data like phonebook, images etc. Very often these applications also require an Internet connection for providing services. Large amount of user data could be gathered eventually using a chat messenger application. It might also have the feature to access his geolocation and notify his friends located nearby. All this data could be continuously accumulated on a cloud. Combining and analyzing the available data could reveal pattern of the user's location preferences. This new knowledge can be further used by the providers to display location based marketing campaigns. Even if a user authorizes the access to data, there remains other issues. Since the entire infrastructure is connected, they are highly prone to thefts and attacks. The cloud should be protected from such malicious intents, since they contain highly sensitive user information. The data in transmission and connected physical objects or devices should be provided with proper security, also. The risk impact could be significant for applications that concern e-Health scenarios. For example, the information gathering vitals from IoT devices used in health applications could reveal that the person is suffering from specific diseases. It could be used for physically attacking a person. The situation could be even more dangerous for examples of devices like remotely programmable pacemaker.

Decentralization and distribution of the system infrastructure is a solution to alleviate such privacy and security issues. Self describing an object could be a plausible way to address privacy, implicitly. Since information related to physical objects are attached to themselves using tags, a user possessing the object can have legitimate control over its data. Access to the data of such objects could be precisioned by the physical range depending on the type of embedded tag. Thus he can physically ensure the security of data. However, the approach could

be made foolproof by storing data in encoded manner, which could be understood by the target applications. Consider a scenario, where an attacker has developed and placed a “smart bomb” in an airport. It is equipped with a RFID reader that can read electronic passports carried by passengers. The bomb is designed to explode when there are at least 25 passengers of nationality ’X’, within its radius. Encoding the information stored in the passport’s embedded tag would make the data appear opaque to such attackers.

2.2.2 The Tangible Data

Using RFID tagging for pervasive computing is already an upcoming trend. This approach differs from the costly alternative where a global infrastructure needs to be setup. While the latter suits better for expensive physical objects, the former is a cheaper solution that can connect petty ones to the digital world. However, our approach uses the RFID technology as an enabler for all kinds of physical objects to participate in local pervasive interactions. The presence of an object is sensed before engaging itself. Pervasive computations are actuated embodying physical activities and contributes with appropriate data stored in their embedded memory. This smart transformation into tangible data takes place by inheriting the properties of RFID tags.

For example consider a pervasive application that weighs objects. The balance consists of an RFID reader as shown in figure 2.2. It can weigh smart objects that contain the weight stored in their tag memory. These objects are sensed when they are kept on the balance. The pervasive application sums up the weight data and displays it, on keeping multiple objects. The e-balance in the figure shows the the total weight of the two objects as 10kg.

Figure 2.2: E-balance using RFID reader

2.2.2.1 Scope of Tangible Data

Self describing objects is one of the contributions in this thesis. Tangible data enables the self description of physical objects, digitally. Our approach proposes describing objects using high level semantic data. This technique would be more straightforward contrary to the traditional ones from the literature. The usual ones in pervasive computing involve fetching low-level sensor data and then analyzing, processing, abstracting them (sometimes remotely) to detect the physical context, activities, situations etc. within a space. While we agree that the traditional one could be powerful for many situations, our proposal would allow to do away with some unnecessary computations and/or communications for detection of simple situations and interactions.

Everyday used objects like food item, furniture, electronic gadget, stationary, equipments etc. would become smarter when coupled with tangible data. Tangible data could be information like:

- status of the object or its properties; could be as simple as one bit status data indicating the object is sold
- contents of the object like the materials used for producing or for its packaging
- if the object is a food item, the information could be nutrition facts (figure 2.3) or an electronic gadget with technical details with rated voltage, frequency, load capacity etc.
- control data concerning a group of objects
- data for linking group of related objects
- last but not the least, information about recycling could be provided.

Figure 2.3: Tangible Data for food items

Consider for example the object in figure 2.4. The keywords describe some of its properties that wouldn't change quite frequently. Suppose sensors and computations such as image processing are used to analyze the properties shown in the figure i.e. boxed shape of the object or it is made of cardboard, which is recyclable or it weighs 78 grams; it would be an extravagant use of resources.

So we propose that similar static properties should attribute as "Tangible Data".

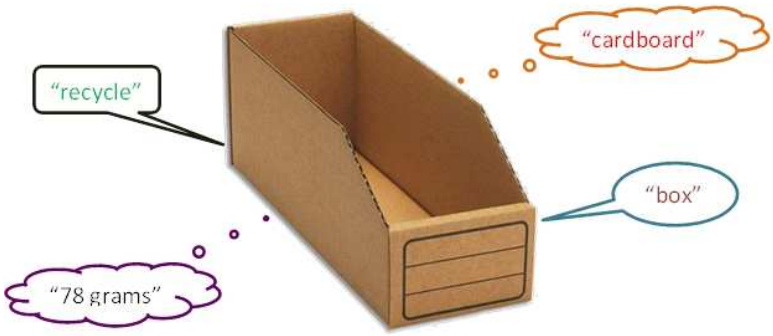

Figure 2.4: Tangible Data for Box

2.2.2.2 Applications using Tangible Data

An object could be described with many properties that would enable to participate and interact in various applications. One might wonder how could this information be useful? The information on our everyday used objects could find use by the retailers, transporters, salesmen at supermarket, consumers, waste recyclers etc. The same piece of information might find use for multiple purposes.

Objects that require to be maintained under specific conditions can have their status indicated as tangible data. Items in cold chains could utilize tangible data. The conditions through which they have traversed can be verified from their status. The data could be checked for anomalies.

For example, consider fragile glass objects have the information "glass" in its description. Everybody handling this specific item in the retail chain would be alerted by the system to be cautious. Additionally, it can be used for recycling process. Trash cans or a glass bin, can accept or reject items selectively based on the materials. This sorting process based on the percentage is elucidated more precisely in the next section. On a massive scale, automatic recycling plants would be able to sort waste items using the information for proper recycling.

A packaging company using robots to screen containers would be able to optimize their choice with the tangible data of each object. Based on the type of parcel for packaging, it can decide on the shape, strength or weight of containers. The robot looking for a cardboard box to package some chocolates would be able to choose the box in figure 2.4. The same box can also participate in automated recycling plants where cardboards are processed.

Objects could be described with unlimited amount of data enabling them for more interactions. For example, if the cardboard box described above is provided with its dimensions, the packaging robot could perform optimally and also cost-effectively. However the box would not be able to participate in a logistic operation when containers are being arranged during transportation. The choice of the describing objects remains upon the users, enterprises, organizations who would want objects to interact with their applications in specific ways. Since the data finds usage across domains, the stakeholders could agree mutually on intersecting sets. Nevertheless the preference of choice should be given to the more generic descriptions that

would enable to take part across various domains; possibly across its entire life cycle, from its manufacturing until recycling.

2.2.2.3 Challenges of Tangible Data

Tangible data is used for describing objects. So, it would be more effective with increasing amount of information. The more an object is vividly described, they would enable greater participation. Since they are required to be stored in their embedded tags, it is currently a challenge from the technological perspective. The tag memory space is quite limited and it requires to be used efficiently. Thus, in the present scenario, a tradeoff needs to be made between expressiveness using data and the memory usage. We have also proposed some ways that would facilitate better expressivity using the available memory.

One of the possible ways is to encode the data elegantly (efficiently expressed) before writing into the embedded tags. The encoding scheme needs to be customized according to the type of tag and the data that needs to be written into it. Hence the self describing information for an object could be application or domain specific. Unlike barcodes and xml data, the information from the tags would be interpreted by machines only. Hence, the encoded data need not require to be human readable. Later in this chapter we have demonstrated using application examples, numeric encoding of object properties. It has better performance, given the nature of application domain. This technique draws motivation from [PB08], which demonstrates efficient encoding of semantic data using prime numbers for resource constraint devices which are commonly used in ubicomp solutions.

Reusing the information used to self described objects is another option to address the challenge. Although it might require collaboration among several organizational entities involved in an object's lifetime. Consider the example of the cardboard box discussed above. It's tangible information could be reused by the logistics as well as the recycling operators. In a way, the process would do away with storing redundant information into the precious memory space of the embedded tags.

However, the challenge could be addressed by various other innovative ways. Combining both above approaches could also be another solution. [EU:] presents a standard document that classifies all wastes and hazardous wastes and are designed to form a consistent waste classification system across the European Union. It lists various categories with numeric representations; "cardboard" is assigned as "200101". This gives an opportunity to combine both approaches in the cardboard box example. For the waste management applications later in this chapter, we have used the numeric categories from the standard document.

2.2.3 Examples

Waste is a menace for the society. If these are not disposed and treated properly, it can be detrimental to the living beings and the environment [Com]. Managing the waste is a huge task, given its ever-increasing volume generated. They could even be complicated at times depending on the nature of waste. They come in many different forms like biodegradable, biomedical, chemical, clinical, commercial, electronic (e-waste), hazardous, industrial, nuclear, sharp, toxic etc. Each of the categories (refer to [EU:]) has to be processed differently. Hence, sorting

must be performed at the earliest for performing appropriate treatment. However, as discussed earlier, some waste contains potentially useful materials for reuse, which are recycled. These need early separation through sorting, to prevent their contamination by other waste types and maximizing the amount of valuable recyclable materials contained in them. Hence, sorting is a very important process for waste management. In this section, we describe two application scenarios for efficient sorting.

2.2.3.1 Selective sorting

As described above, one of the aims for sorting is maximizing the amount of recyclable materials like paper, glass, plastic etc. We present an application which would enable this aspect of maximization. The application processes smart waste containing the information about the amount of recyclables that could be recovered from them. Their embedded tags are encrypted with the recyclable material classification type identifier and percentage information; some tangible data that helps items to participate in the recycling process.

Consider the examples of smart waste in figure 2.5. The cardboard box, glass bottle and plastic bottle are made of recyclable materials. Each of their tags contain the information that they are made of cardboard, glass and plastic with 48%, 97% and 83% respectively. This kind of information would benefit in taking preferential decisions when sorting. Suppose a sorting process wants to gather glass (type identifier 200102) waste with at least 85% purity. Among the three items in figure, the glass bottle in the centre satisfies the conditions in respect of both, material and its quality. Hence, it would not be possible to contaminate a sorting process with inferior or other materials. As a matter of fact, contamination could also be the other way; like, adding the bottle from above example for a sorting process collecting low quality glass, between 40% and 75%. So, it is up to the sorting process in choosing the purity range (in %) of recyclable materials while collecting waste.

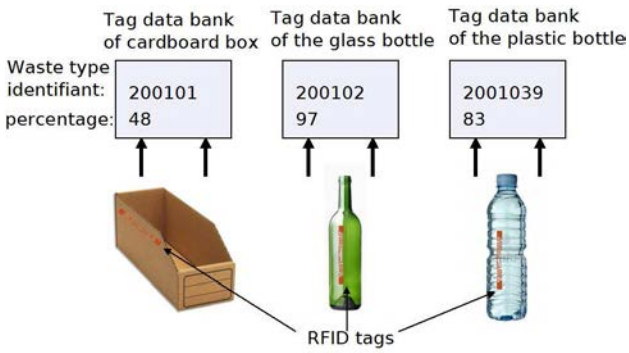

Figure 2.5: Representation of recyclable material percentage as Tangible Data

Practically there are various forms of waste that contain multiple materials. A typical example of such form would be e-waste or electronic waste. Computers, telephones, televisions, etc.; all such electronic items contain recyclables and hazardous materials. Plastic, glass, metal are some that can be extracted for reuse. Hence for such forms of waste, the above sorting

process requires some modifications to its conditions. The sorting conditions must have flexibility to accept items containing multiple recyclable materials; unlike the process explained in the preceding paragraph. Consider for example a sorting process that accepts waste items containing glass (identifier 200102) $<$ 35% and plastic (identifier 2001039) $>$50%. The two conditions are represented by the two coloured circles in figure 2.6 with their combined at the intersection. The monitor in the figure, is tagged containing composition information of type and quantity of materials used i.e. 25% and 55% of glass and plastic respectively. The composition satisfies the conditions set for the sorting conditions and hence would be accepted.

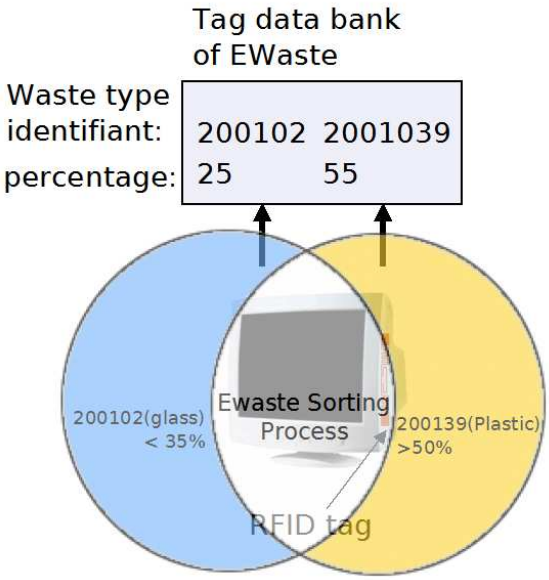

Figure 2.6: Representation of recyclable material percentage as Tangible Data

2.2.3.2 Detecting interactions

In the preceding example, we explained how an item's self-description information with its quantitative measures (like weight or percentage) of recyclable materials could enable efficient sorting. The smart bin is a collective container that has an embedded computing system to read and process the tangible data. However, there are other ways to perform sorting for better waste management. The objective is to maximize on the concentration of valuable recyclable materials. An accumulation could be contaminated due to the presence of other particular materials. This could render the entire collection unfit for recycling. Consider for example a glass bottle put into a paper or plastic bin. This would reduce the recycled value of the collected paper or plastic items [onl]. Apart from mixing of materials, there could be physical hazards reducing the value of collected items. A flame caused due to an explosion from aerosol can in high temperatures could ruin the collected paper or plastic materials. Disposal of such unsafe items in the same waste bin could result in a snowball effect of physical hazards. The remaining part of this subsection provides a briefing on the proposed system design before explaining the

working principle that enables avoiding such linked incompatibilities.

System Design

In this section, we describe the system design for inferring incompatibilities locally. It essentially means that all the information required is available without referring to remote database or knowledge base. An alternative could be to distribute the information partially among the waste items and a local knowledge base. The waste items are identified by the system before inferring on incompatibilities. We have chosen a commonly used architecture (for sensing and actuation) for our system, as shown in Figure 2.7 below.

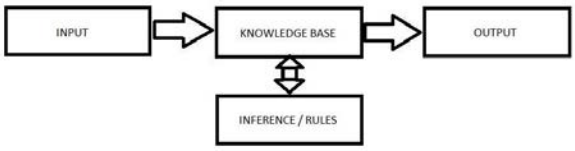

Figure 2.7: Commonly used Architecture for Systems

We describe the components briefly.

- Input: It is that point in the system where the waste items are identified and added.
- These two components (using OWL ontology, explained later) are embedded in the smart bin
  - Knowledge Base (KB): This contains all the required information to identify the items along with their properties. It also updates its knowledge regarding the presence of items that are being added to the system incrementally.
  - Inference/Rules: This component of the model uses the KB to reason out about the possible incompatibilities and hazards. The inferences are added back to the KB.
- Output: It sends out notifications to communicate about alerts and warnings to the users of the system.

Principle

Smart waste items have self description of their properties using RFID tags. Based on their properties which constitute as tangible data, incompatibilities are computed among a collection of items present locally. In this section, we discuss its underlying principle. For the purpose we begin with organizing the waste domain in a specific manner for making such inferences.

Describing waste items The waste domain can be categorized based on their various hazardous properties. There are standards that specify the properties of waste materials and categorizes them [EU:]. It discusses the domain in great details. We have preferred to reuse this existing knowledge for our purpose of self describing the waste items. Our approach reuses the categorization and demonstrate few examples of hazards related to some of these categories.

Some examples of hazardous properties for this domain are spark, explosion, toxic fumes etc and can categorize based on them. As discussed in the previous section, we are interested

to infer incompatibilities. So, the essential properties are picked that are only relevant for interactions with other items. We have proposed a "prolog like" system that computes the interactions using first order logic. The encoded self description of the items present locally are combined to form the knowledgebase and then reasoned for the interactions i.e. incompatible relations among categories. We have used OWL (Web Ontology Language) for constructing the local knowledgebase. Since OWL uses the open world assumption, it was the preferred choice over other first order logic languages. "The closed world assumption implies that everything we don't know is false, while the open world assumption states that everything we don't know is undefined." [2] It implies that our smart bin with OWL reasoner would display an alert when it is sure of an incompatible interaction using the knowledgebase. For the remaining cases, it's interpretation would be $\mathrm{not\ computable}$. Thus the later would avoid false negatives.

Figure 2.8 represents pictorially the data structure model used for describing waste categories. Its individual fields are described as follows:

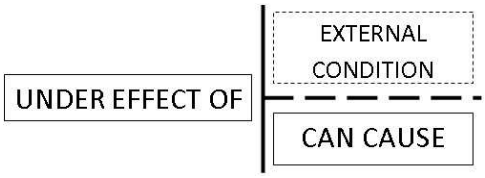

Figure 2.8: Conditions to describe a category

- under effect of: the condition(s) which holds the properties that can influence the category
- can cause: this condition enlists the hazardous properties that the category is capable of causing
- in presence of: this holds the external conditions under which the $\mathrm{can\ cause}$ properties occur; they are the physical environmental conditions that need to be captured using sensors.

In the subsequent sections, we will use the same pictorial representation to describe the waste categories or items in our examples.

Let us take some scenarios of interactions between categories. First, let's take an example of simple incompatibility between a pair of them. Suppose a category A can cause an incidence (for instance say hazardous property X) that affects a second category B. Hence, an incompatibility exists between the categories A and B. Our second example is a slightly more complex and realistic than the previous example. If the category A causes the incidence (i.e. X) only in presence of an external condition (let's name as C), makes it an important augmentation to the scenario. Hence, the categories do not pose to be incompatible if the condition C is unfavorable. Both of these scenarios consider the incompatibility between different categories where the hazardous property affects each other. However, there are properties like explosion for example, which have hazardous effect by itself. The situation can be represented as a category

2. -Stefano Mazzocchi, "Closed World vs. Open World: the First Semantic Web Battle", http://www.betaversion.org/ stefano/linotype/news/91/

that causes a hazardous property that affects itself and may depend on the external condition.

Inferring incompatibilities As described above, we can self describe waste items accordingly. When a collection of these items is present locally we can infer incompatibilities based on the discussed scenarios. Sometimes objects are located remotely and communicate within themselves and other knowledge base using network infrastructure like the Internet to make decisions. Such an idea is called Internet of things (IoT) in the field of pervasive computing. Our approach here, proposes making the required information that describes waste domain available locally for inferences. Such collective inferences could be made without using network for communication. We prefer to use the name for such a situation as Intranet of Things (InoT) as it does not involve any items located remotely and differentiate to avoid confusion.

We discussed above, the interaction scenarios between pairs of categories based on hazardous properties. Multiple such categories can constitute an InoT. The graph in Figure 2.9 represents an example of InoT formed. The shaded nodes represent some categories. They are connected by an edge if they interact. The dotted edges represent interactions which are unfavorable due to external conditions. One of the external conditions were high temperature at the instance this snapshot was drawn. Hence, the dotted edge encircled in the figure representing an interaction under low temperature becomes unfavorable. The firm edges represents favorable interactions, which could be either the first or second scenario described above. The shaded node with a self-loop represents the last scenario, which is favorable in this case as the external condition is satisfied.

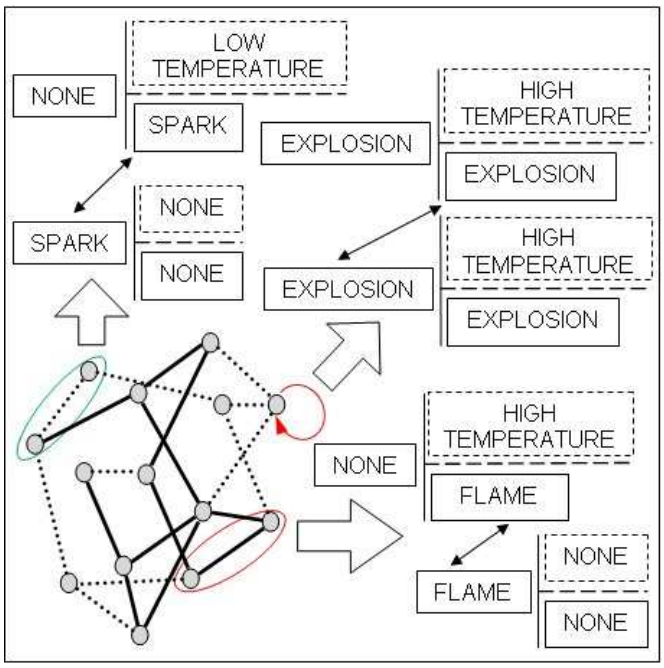

Figure 2.9: InoT formed

Finally, if a waste item belongs to one or more categories, it would contain all their conditions. Hence, they could be used for collective inferences also.

Annexure A contains further elaborated description on how the system is designed based on the architecture and reasons using the principle discussed earlier in this section.

2.3 Coupling Associated Objects

Tangible data is an innovative way for self description of objects. We have already discussed the approach's advantages; which also addresses the drawbacks of centralized approach using RFID. However, the challenge exists with the available memory. Although this restriction would fade away in the years to come i.e. the tag memory would increase and get cheaper in years to come. To make the best out of the current situation, we propose utilizing the memories of multiple tags. If embedding a single tag does not suffice for storing the tangible data, multiple tags can be used to provide the necessary supplementary memory space. This is the simplest, quickest and cheapest solution as tags can be bought for few cents. Alternatively, the information could be distributed over multiple object tags. Generally, objects are in groups. In the remaining of this section, we explained this approach in details and demonstrate various applications using examples.

In our everyday life, we come across very often group of related objects or a complex object made of many parts. If each of these objects are tagged, their combined memory would be higher to store tangible data with more precision. Structuring and distributing this data in a fitting way would enable customized pervasive services like arranging, grouping, verifying missing objects etc. The objects would not only contain self description but also information about its related ones. Hence in this way their relation in the physical world could be represented digitally using RFID tags. Take for example an arrangement on the bookshelf. If the tangible data on each object includes information about its adjacent book, the arrangement could be represented digitally. Figure C.4 illustrates the approach where self description of objects includes its related ones. The digital information linking to adjacent books are shown using arrow lines.

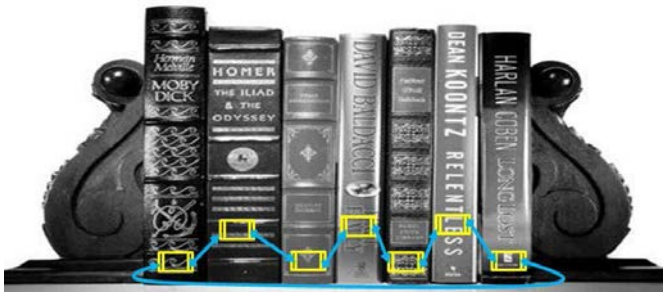

Figure 2.10: Bookshelf with related objects (books)

A mobile phone could be considered as a complex object as shown in figure 2.11. Suppose its most important (and costly) parts have embedded tags. So instead of storing the tangible data

of the mobile phone into only one tag, we propose an approach to distribute information over multiple tags. This allows a balanced usage of the tag memories as well as offers additional services like verifying the integrity of the mobile phone. The information is distributed in such a way that the tangible data of the parts have a reference among themselves. These references could be in form of some data structures like linked list, graph or tree depending on the natural association among the physical parts. Hence the name Tangible Data Structure is derived. The logical association of the physical group is called coupled objects. Figure C.5 shows the mobile phone as a coupled object associated using a graph data structure. Composite objects that require step-by-step assembly could be represented as a tree structure. Figure C.6 demonstrates an hierarchical representation of a chair as composite object using its parts. Both examples of complex objects are used later for detailed explanation of Tangible Data Structures.

Figure 2.11: Complex Object

2.3.1 Our Tangible Data Structure for describing related objects

Definition 2.1 (Graph) A graph G is a pair (V, E) comprising a set V of vertices, and a set E of edges, which is a set of pairs of vertices belonging to V.

A tangible data structure consists of a group of tags modelled by a graph. Each of its vertex is represented by a RFID tag. Each tag stores in its memory tangible about the neighbors: for example their tag id or a nickname, to address them. Each tag knows the nickname of its neighbor in the graph representation. These data are used to store the edges of the graph representation in the tag memory of the group.

Definition 2.2 (Complete Graph) A complete graph is a simple undirected graph in which every pair of distinct vertices is connected by a unique edge.

If the graph is completely connected (see figure C.5(a)), it would be possible to verify the integrity of the coupled object. The structure is maximally robust such that the information

Figure 2.12: Composite representation of a chair as hierarchy

from one part can also be used to trace all the remaining ones. Figure C.5(b) show examples of the data stored in PCBs and LEDs embedded tags. Using the tangible data of PCB, we can retrieve the list of missing parts. The same holds for LED or KPAD or LCD. The advantage of the approach is its autonomy. However, as the size of coupled object grows, we are faced with the challenge of the tag memory size; since the amount of tangible information also grows proportionally and equally for each of the tags. Alternatively, regular graph provides a balanced optimum solution that consumes lesser tag memory but compromises on the robustness of the tangible data structure.

Definition 2.3 (Regular graph) A regular graph is a graph where each vertex has the same number of neighbors: every vertex has the same degree. Let $k \in \mathbb{N}$. A regular graph with vertices of degree k is called a k-regular graph or regular graph of degree k.

Using a k-regular graph ensures that the same amount of memory is used in each tag. This is because every node (representing a part of the coupled object) is connected to k other nodes (as per definition of Regular Graph). It also means that the existence of every node is replicated exactly k times; thus determines robustness. Hence a missing node can be traced through any of its k connected neighbours. But, considering the entire k-regular graph, it is possible to trace at most k nodes [GC12]. This implies:

- when a set of n nodes are lost, with $n \in \mathbb{N}, n \leq k$, it is possible to trace all the n missing tags with their known nicknames/identifiers.
- each among the k neighbors that are present, provide the information about the missing nodes.

Figure 2.14 shows our previous example using a 2-regular graph. Consider the following possibilities for the coupled object:

- if LCD goes missing, it can be traced from the tangible data stored in KPAD or PCB.

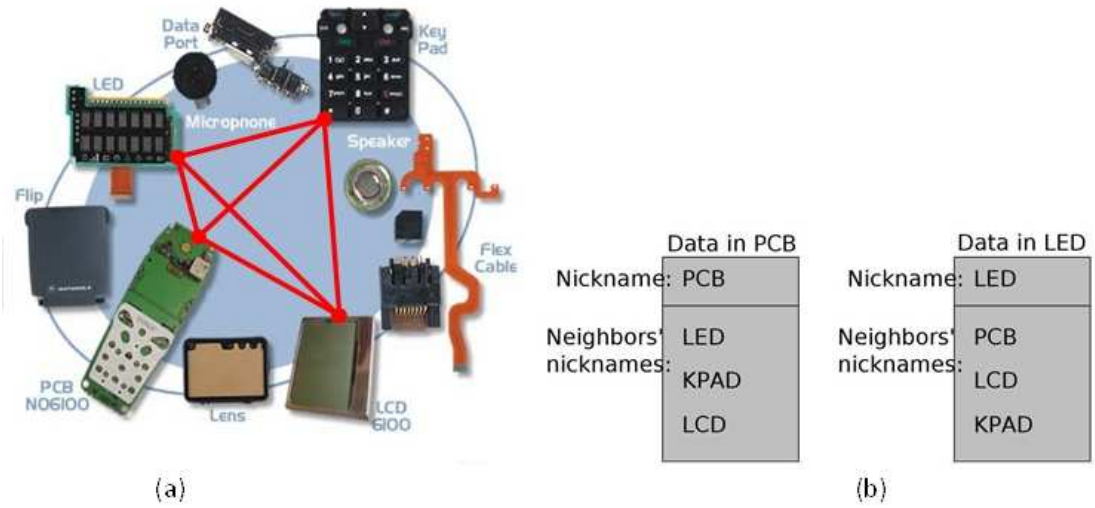

Figure 2.13: Coupled Object using Complete Graph

- if LCD and KPAD goes missing, they can be traced from the tangible data stored in PCB and LED respectively.
- if LCD, KPAD and PCB goes missing, then
  - KPAD and PCB can be traced from the tangible data stored in LED.
  - LCD cannot be traced because both its neighbours i.e. KPAD and PCB are also missing.

This shows, that it is possible to trace up to 2 items in the coupled object, using a 2-regular graph in the tangible data structure.

UbInspect: Application identifying missing objects

Another simple application scenario of coupled objects is Ubi-Check, which notifies for missing elements [BAC09]. UnInspect presents a robust form of the same. It notifies as well as identifies the missing elements among coupled objects. We would illustrate the UbInspect application in this section.

Consider someone at the airport who is going to cross the security gate. The person is required to remove his jacket, belt, and put in a tray mobile phone, music player, remove laptop from his bag, and may be other objects. All that in hurry, with other people in the queue doing the same. Obviously, personal objects are vulnerable to get lost in this situation: objects can get stuck inside the scanner, can stack up on each other at the exit of the scanner. And so, it is easy to forget something while being stressed to catch a flight in hurry. A coupled objects solution would secure the process by associating all personal items into a group, and checking it at the exit gate of the screening area. The same application could also work when leaving an hotel, a train, a plane, etc. The Ubi-Check scenario is used in [GC12] to demonstrate the robustness

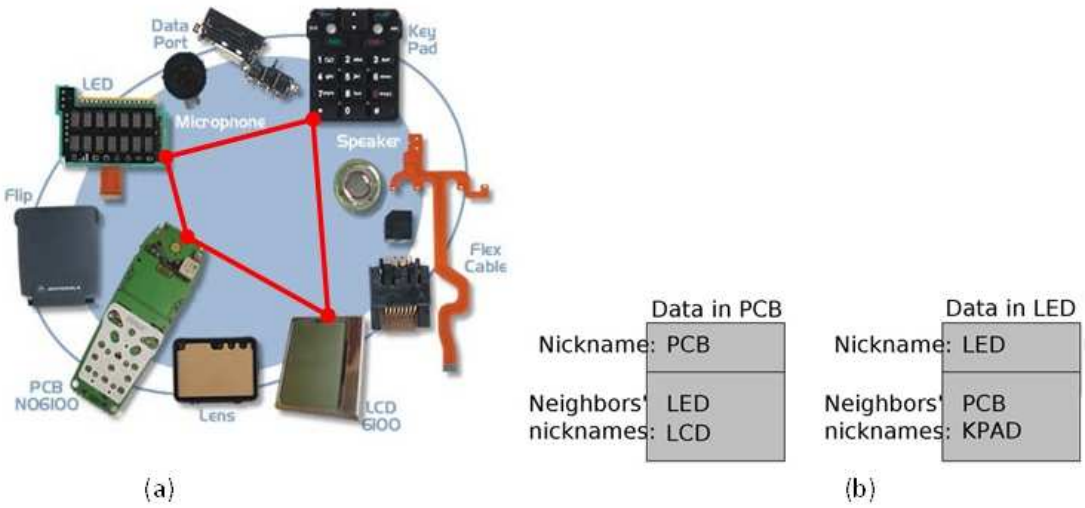

Figure 2.14: Coupled Object using 2-Regular Graph

of k-regular graph structure and memory cost. A traveler have 8 objects: Phone, Wallet, Bag, Belt, Jacket, Passport, Watch, Laptop, that he considers as very important. He decides to couple these objects with the Ubicheck application based on a RFID solution. Figure 2.15 shows two instances of coupling using 3-regular graph and 4-regular graph. The first and and the second cannot trace the above 3 and 4 missing objects, respectively. Figure 2.16 (a) has 4 objects missing among which passport cannot be traced from data available from the present items. Similarly, figure 2.16 (b) uses a 4-regular graph and it would be unable to have any clue about the missing jacket among the missing objects from the information present. The missing objects that could be traced in figure 2.16 are marked with red.

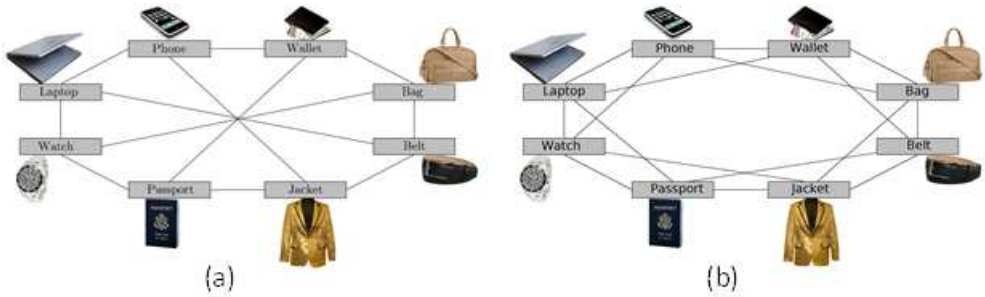

Figure 2.15: UbInspect using (a) 3-Regular Graph (b) 4-Regular Graph [GC12]

The characters can be encoded in a hexadecimal value on two bytes. Let us consider, that the nicknames have at most 10 characters. In a 3-regular graph representation, each tag must store the three nicknames of these three neighbors. Then, it is necessary to use 60 bytes to store

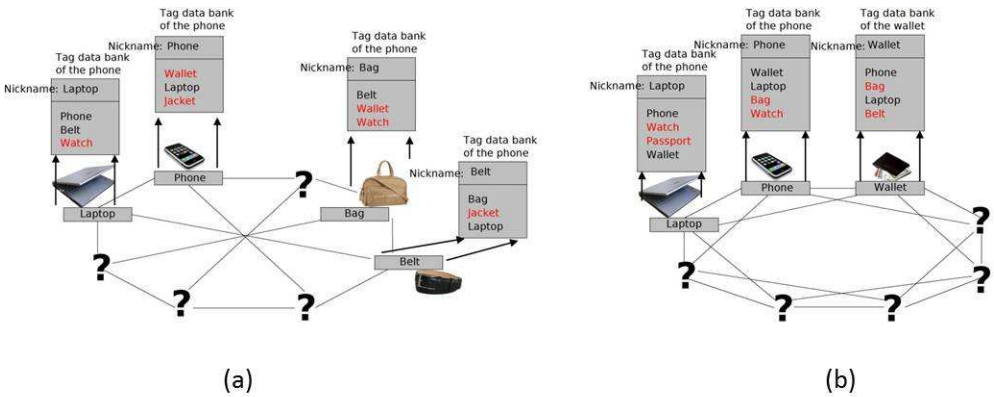

Figure 2.16: Integrity check fails: (a) 4 missing (b) 5 missing Objects [GC12]

the nickname of the neighbors in 2-regular graph. Typically, RFID tags contain memory sizes from few hundred bytes to kilo bytes. More generally, for a $k-$regular graph representation, when the maximum length (in number of characters) of the nickname is equal to l, the memory space used to store the nickname of the neighbors of a tag is equal to: $2 * l * k$. Here, each tag is identified by its nickname which is a good way to extend the Ubi-check application.

The above examples demonstrate that k-regular graph as the tangible data structure enable to verify the integrity and detect the missing parts autonomously, using their embedded tags. Although it is not as robust as a completely connected graph, yet an optimum can be decided based on requirements by adjusting the value of k. The self description of the complex object is distributed over several tags embedded into its various parts. The digital information is attached to the physical objects and not stored in remote information system. Hence privacy is also addressed.

Apart from verifying the integrity of coupled objects, Tangible Data Structure could also be utilized for providing services like assistance, autonomously. However, the graph structure should have more flexibility to represent physical complex objects naturally. In the next example, we elucidate the use of a labeled (or vertex-labeled) graph like representing the arrangement of tools in a toolbox, that can be inquired for missing tools at various checkpoints of sensitive sites. Then move on to hierarchical structure i.e. tree and explain how they could be used to assist in assembling physical objects.

2.3.2 Vertex-labeled Graph

Definition 2.4 (Vertex-labeled graph) It is a graph where each vertex is labeled (traditionally represented by integers) in addition to the data that identifies the vertex.

In our previous approach, we have explained the use of regular graphs for Tangible Data Structure. Since each of their nodes have equal and fixed number of neighbours, the upper bound of the memory consumption is known. However, the situation is different when using

vertex-labeled graph. It allows the flexibility on associating the parts to form a coupled object depending on their importance and embedded tag memory size.

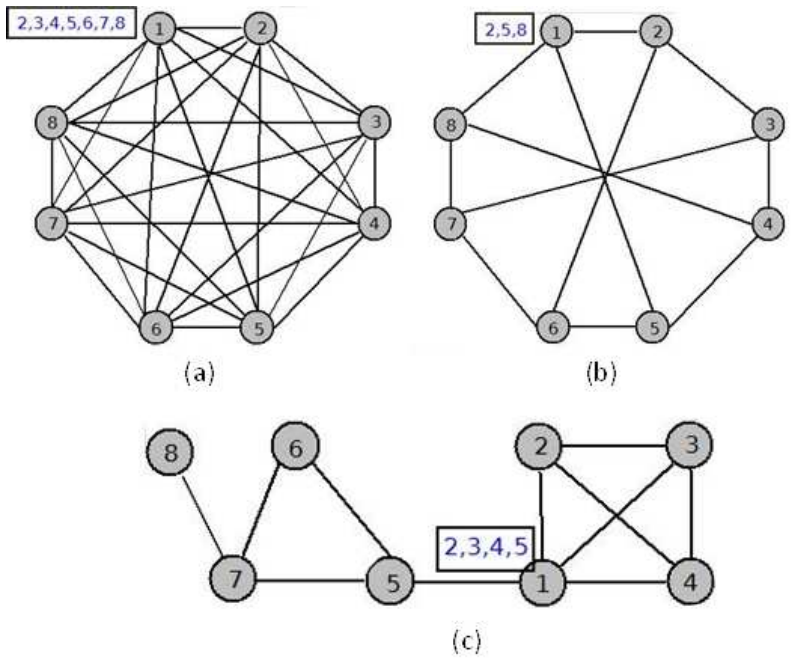

Figure 2.17: Neighbours of Node 1 in (a) Fully Connected (b) 3-Regular (c) Vertex-labeled Graph

Figure 2.17 compares three different structures containing 8 nodes. Figures 2.17 (a) and 2.17 (b) show their nodes have 7 and 3 neighbours respectively. But, the nodes in 2.17 (c) have unequal number of neighbours. While node 8 has only one neighbour, node 1 has four neighbours; which is also the highest in the graph. So its nodes have varying amount of information. A node can contain neighbours from one up to a finite number. The graph can also contain cliques (or complete sub-graphs). The node sets 1,2,3,4 and 5,6,7 are cliques joined together by nodes 5 and 1. Nodes belonging to a clique would contain more information, having more neighbours. Hence these part(s) of the graph would be more robust.

Vertex-labeled graph would allow to build a customized coupled object. The important (or costly) items in the complex object can contain better tags with more memory space. So they could be made part of clique(s). Conversely, physical items embedded with more memory space should represent nodes belonging to clique in the tangible data structure. Summarizing the advantages of using vertex-labeled graph:

– link between neighbouring nodes vary from being pairwise to fully connected.
– nodes have varying amount of information; hence supports variety of tag memory space.

2.3.2.1 The Smart Tool Box Application

The Smart Tool Box is an example of ubiquitous application which helps to maintain safety standards at sensitive sites like aircraft [RSMD04]. Let's consider a workshop has a depot for tools which issues them to its workers in a toolbox. Each type of tool in the depot is assigned an unique identification. Initially, every tool is RFID tagged which stores its type id. When a worker requests for some tools, they are lent out in a tool box. Henceforth, the box should always contain all the tools grouped together and warn the user if one or more tools are missing until returned back to the depot. The vertex-labeled graph based tangible data structure is computed and the information is written onto every appropriate tag before handing out the kit to the worker.

We use the tangible data consisting of tool type identifiers for integrity checking. In the present context, the domain consists of tools used for maintenance. In the following subsections, we have illustrated how the domain knowledge i.e. tool identifier could be represented efficiently to be used to perform grouping of tools at the depot, on request. Our objective is achieved and validated using a simple android application. Finally, we discuss the effectiveness, including an evaluation of the encoding used to write the tangible data.

Tangible Data representation for the Tools Domain

From the above discussion of vertex-labeled graph, we can speculate that cliques are interesting in the graph. So, we intend to keep the clique information separate from the pairwise association of nodes; considering interesting utilities in future. Also the encoding should be efficient in the RFID tag for self describing each tool. We have proposed to construct the tangible data for a tool $\mathbb{T}$ in the following string format:

$$f_0{:}f_1{:}f_2{:}f_3{:}\ \dots\ {:}f_n$$

The string consists of n fields with colon as separators with each containing a set of type identifiers. The first field f_0 contains the type identification of $\mathbb{T}$. The second field f_1 contains the type id of all tools pairwise associated with $\mathbb{T}$. The various cliques with which $\mathbb{T}$ is associated are encoded in the fields from f_2 to f_n. Provision for these multiple fields would add the flexibility of having the tool $\mathbb{T}$ associated with more than one disjoint cliques. From the example in figure 2.17 (c), a tool $\mathbb{T}$ represented by the node 1, can self describe itself as:

$$1\ :\ 5\ :\ 2{,}3{,}4$$

Let us consider the toolbox used for our implementation. We have used numeric representation of tool type written on their NFC tag in the format described above. So each type of tool performing distinct function in the depot is assigned a unique natural number. Suppose the depot lends out specific type of toolbox with two compartments as attachment. They are assigned type identifier 1, 2, and 3 respectively and represented as a clique as shown in figure 2.18 (a). Their tangible data is updated in their field f_2 as shown. Then tools of type id 7 and 8 are added to compartment 3 of the toolbox. And a chisel of type id 14 is added to compartment 1 of the toolbox. These additions of the tools are considered as pairwise associations with their

respective toolbox compartment and all the related tags are updated. Figure 2.18 (b) shows graph representing the arrangement of tools into various compartments of the toolbox and the tangible data for the tags.

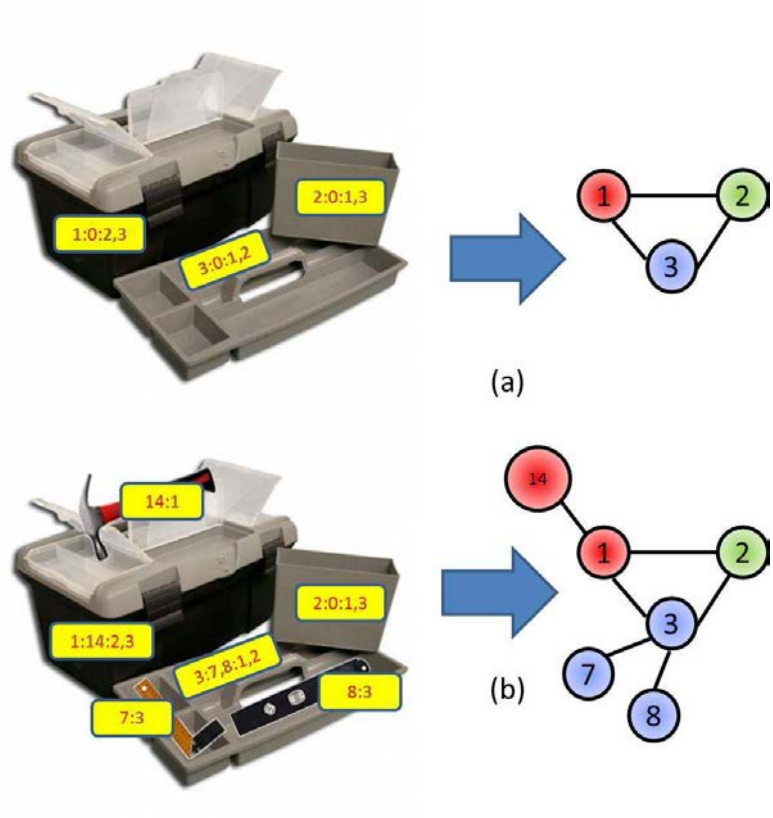

Figure 2.18: (a) Toolbox as clique (b) Adding tools to different compartments [SC13a]

However, before writing the final data onto the tags, we perform a tweak for memory efficiency. Every field of the string format is added with a number specifying the fixed length of the type identifiers. Also numeric identifiers are padded with '0' on the left to make them all of equal length. According to the proposed format, the data for toolbox with id 1 will be modified to:

$$11 : 214 : 123$$

If a field is empty, it contains 0 which represents properties of 0 length i.e. another way of saying that no property exists for the field.

Validation using Application

Algorithm 1 outlines the procedure to to verify the integrity of the toolbox. While the pairwise associations are verified individually, the tools that are in a clique requires to be present together.

Algorithm 1: Inferring integrity of toolbox

```
Input: tool 𝕋 detected by the NFC reader
Output: infer if all the tools associated with 𝕋 are present locally
Initialize: Set S ← ∅
while tool 𝕋 detected by the NFC reader do
    set Flag to TRUE i.e. assuming all the tools associated with 𝕋 are present locally
    Read field f_1 from the tag
    for each tool id p in field f_1 do
        if p ∈ S then
            set Flag to FALSE i.e. tool p pairwise associated to 𝕋 is missing
    for each field f_i from f_2 to f_n do
        set GroupFlag to TRUE i.e. assuming all the tools associated with 𝕋 as clique in
        f_i are present locally
        for each tool id Ip in f_i do
            if Ip ∉ S then
                set GroupFlag to FALSE i.e. tool with id Ip in the clique f_i is missing
        if GroupFlag is FALSE then
            set Flag to FALSE i.e. some tool in the clique f_i is missing
    if Flag is TRUE then
        for each element p in Prop_i[] do
            S ← S ⋃ p i.e. all associated tools with 𝕋 are present. so adding the id of
            tool 𝕋 to set S
    return Flag
```

We have developed a small android application to validate and verify the working of our current example. Figure 2.19 demonstrates the various phases of the application. It shows an instance of verification being performed on a set of tools. During this process, every tool is scanned to read the self description in the NFC tag. The tool type is read from the field f_0 of the description and its presence is asserted as in figure 2.19(c) when tool type 26 is added. The group information is extracted from field f_2 and the presence of each tool type is inquired. Figure 2.19(d) notifies the user about missing tool type 9.

Encoding Efficiency The length of each field in the self description would depend on two factors. The maximum length among the group of numbers in the particular field are chosen as fixed length. So each number having smaller lengths are padded on their left adding to the overall length of the field. However, if the maximum length among the numbers is less, then

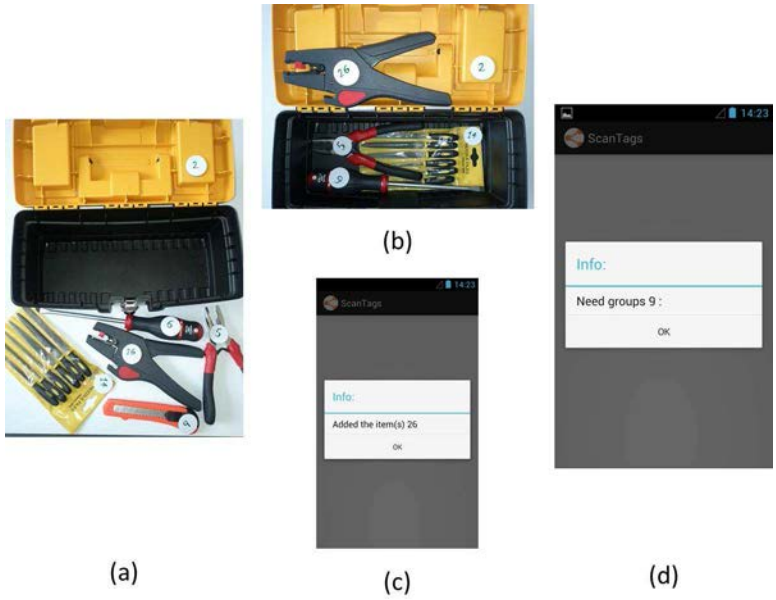

Figure 2.19: Smart Tool Box Application (a) Initial grouping of Tool Box and contents (b) Integrity Verification of the Tool Box (c) Acknowledgement on adding tool numbered 26 (d) Notification for tool(s) yet missing

the encoded string automatically gets reduced.

Suppose we have the following sets: A domain $D = \{x \epsilon N\}$ and $|D| = n$, where n is the number of properties represented.
Any field in the self-description of an item could be represented as set $F \subset D$ ...(i)

Let's define the following functions for some operations:
$|X|$ gives the cardinality or size of $set\ X$.
$max(X)$ gives the largest element of $set\ X$.
$len(element\ e)$ gives the character count of element e.

We can say that,
$max(F) \leq max(D)$
or, $len(max(F)) \leq len(max(D))$ [This gives the character count of highest elements of both sets] ...(ii)

Now, if we represent the field in (i) using the fixed length, we can write
$|F| \ * \ len(max(F)) \leq |F| \ * \ len(max(D))$ ...(iii)

Hence, we can make three observations for the encoded length of the field which is the L.H.S of the equation (iii):

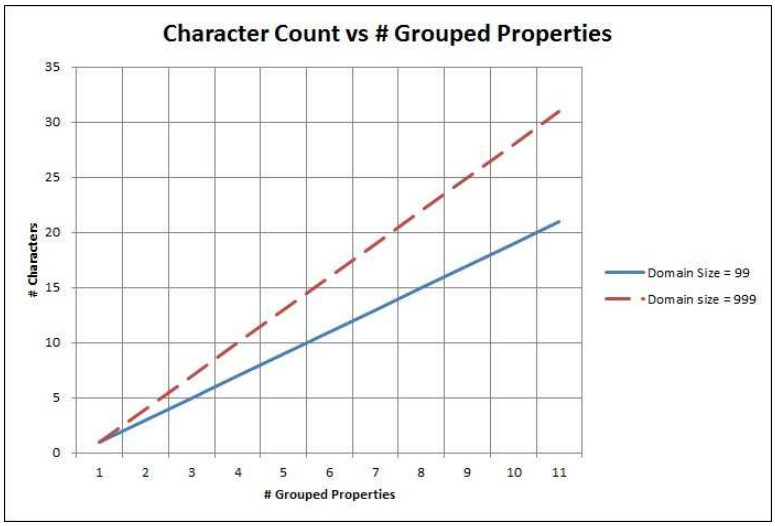

Figure 2.20: Encoding performance

1. It depends on the number of properties grouped together represented as $|F|$.
2. The length of the largest number also contributes to the total length.
3. The maximum length is bounded up to the R.H.S of the equation.

Figure 2.20, provides an estimate on the upper bound of the characters required for encoding groups of properties in a field depending on the domain size. The maximum length of the properties for the domain are considered in the graph and hence the lines outline the upper bound on the total length of the field which can never be exceeded. However there may be circumstances when the total length is reduced if the group of numbers in the field comprises of smaller numbers.

For domains having average number of properties, they can be represented by the natural numbers until 99 having 2 characters as fixed length in the fields. For bigger domains with more properties, larger numbers can be used. But using natural numbers upto 999 would probably be sufficient for most domains as it has large interval of numbers. Also the encoding length is guaranteed to remain within the upper bound for the entire interval given the number of properties for a group.

The entire encoded string data to self describe an item contains numbers and the special character ':'. When the data is written onto RFID tags, we replace the colon with character 'a' and use 'b' as the data delimiter. Hence the string is transformed to hexadecimal and each of its character consumes four bits of memory. Hence using numeric properties gives us an advantage whereas in some cases it could even take upto two bytes for a character [GC12].

Advantages : Deployment, Scalability and Privacy The entire setup is very easy to build. It is effective in terms of cost and time. Initially, it would require identification of the properties of the domain i.e. different type of tools and assigning them unique identifier using cheap NFC tags. The interoperability of the tools among multiple deployments, spatially, is possible as

long as their properties have the same significance. The grouping could be done later on-demand, based on the requirement of a worker. Reading and writing of the NFC tags are performed with applications using an android phone. When a large number of tools are grouped together, scalability factor comes into play in two different ways. Firstly, the memory space of a tag could be too limited for its self-description. This is resolved by sticking multiple tags instead of one. Alternatively, the tool with limited constraint could also have pairwise relation with another within the data structure. Secondly, the information is available locally for the entire setup. So, the absence of a network reduces the communication time and the only time taken is for the read/write of tags, performed one at a time. Also, this addresses the concern for privacy as the entire information is available locally instead of a server. When information are fetched from servers, the access patterns are sometimes recorded for profiling which is not possible in our framework. Last but not the least, this approach is simple, energy efficient due to use of passive tags and have moderate user interaction compared to the approach in [AMS02].

2.3.3 Distributed Tree Structure

We perceive and use hundreds of physical entities around us in our daily lives. While some of them are simple, the rest are complex ones that are built progressively out of simple ones. Such types of complex objects could be represented naturally in form of hierarchy. Other examples of hierarchical representations exist in science, technology, language, information classifications etc but we intend to focus on areas of application where small entities are used as building blocks towards larger ones. Few examples of such complex physical objects are assembly of electronic equipments, vehicles, ready to use furniture, data blocks in file system etc. Activities such as assembling and reassembling during repair often require to check for the integrity of such complex objects and caution for missing pieces.

We propose hierarchical representation of complex objects distributed among its real physical entities as tangible data. Augmented with such information, these entities become smarter to take part in interactions like assembly etc. The next section describes our data structure for the hierarchy. In a later section, we have explained two algorithms. The first, disseminates tree structure information onto the objects. Henceforth, they can participate in assembly to reconstruct the hierarchy using the second algorithm.

2.3.3.1 Data structure for hierarchical objects

Let's suppose, we have a physically assembled complex object using multiple elementary objects and define it as a tree structure based on the following design principles. The most elementary physical objects would be represented as leaf nodes. All internal nodes should have at least two child nodes since they represent an association of two or more parts. These children could be leaf and/or another internal node. The internal nodes virtually represent a complex object composed of elementary physical and/or other complex object(s). Hence, traversing a tree bottom-up, would resemble the assembly process, stepwise. On the other hand, top-down traversal of the tree disassembles the complex structure with the root node modeling the complete structure. Figure 2.21 presents an example of a complex object using hierarchical

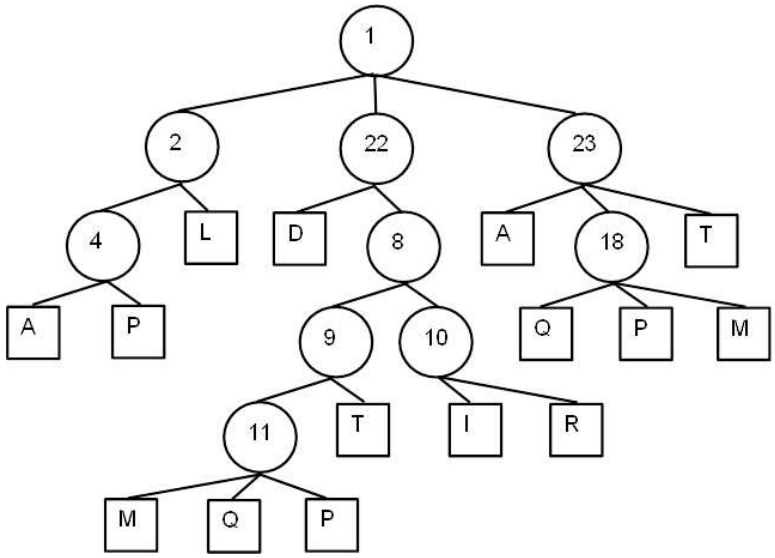

Figure 2.21: Sample tree structure to be distributed among leaf nodes

structure. Its set of leaves marked as boxes are the physical parts while the remaining in circles are internal nodes representing complex objects.

Starting from this tree structure, we define the notion of parent, ancestor, and sibling of a node n belonging to a tree $\mathcal{T}$.

Definition 2.5 (parent) Let a tree $\mathcal{T}$ and a node n belong to $\mathcal{T}$. The parent p of n is the node of $\mathcal{T}$ for which n is a child of p. The set of children of a node p in a tree $\mathcal{T}$ is denoted by $\mathcal{C}_{\mathcal{T}}(p)$.

Definition 2.6 (sibling) Let a tree $\mathcal{T}$ and a node n belong to $\mathcal{T}$. The sibling of n is the set of nodes that have the same parent node as n and that are different nodes from n.

Definition 2.7 (ancestor) Let a tree $\mathcal{T}$ and a node n belong to $\mathcal{T}$. An ancestor a of n is a node for which n belongs to one of the sub-tree that have a as root node. The set of ancestors of a node n is denoted by $\mathcal{A}(n)$

Considering the tree example presented in Figure 2.21, the node 1 is its root node. The nodes in boxes marked with letters are leaf nodes. The node 4 is the parent node of leaves A and P whereas nodes 1, 2, 4 are their ancestors in the tree.

Definition 2.8 (Internal nodes) Let a tree $\mathcal{T}$. The set of internal nodes of $\mathcal{T}$ is the set of nodes a of $\mathcal{T}$ such that a is not a leaf of $\mathcal{T}$. The set of internal nodes of $\mathcal{T}$ is denoted by $I_{\mathcal{T}}$.

For a tree $\mathcal{T}$, $I_{\mathcal{T}}$ is the set of nodes that are not a leaf. In Figure 2.21, all the nodes marked as circles are internal nodes of the tree.

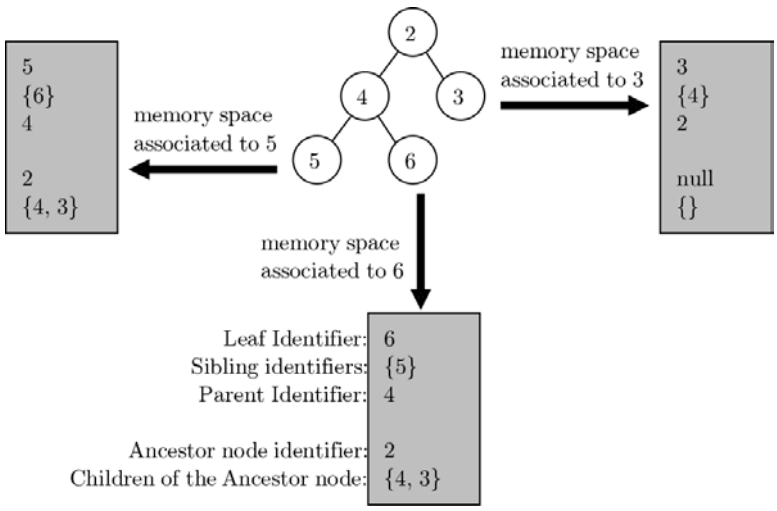

Figure 2.22: Memory space organization of the leaves

Our design approach of the hierarchical objects, aims to attach the digital information about tree structure to the fundamental parts i.e. physical leaves. As presented in Figure 2.22, each leaf is associated to three types of digital information:

– the first type of information describes the elementary part: the leaf of the tree,
– the second type of information describes the parent of the elementary part,
– the third type describes an ancestor node of the elementary part.

The data about internal nodes are stored in the memory space attached to the leaf, representing a physical object. Hence it should be ensured that enough space is available to store the tree structure information to the memory associated to the physical leaves. This is ensured by the following property proving by induction that the number of leaves always exceeds ancestor nodes in a tree, independent of its depth. The set of leafs of a tree $\mathcal{T}$ is denoted by $\mathcal{L}_{\mathcal{T}}$.

Property 2.1 Let a tree $\mathcal{T}$ such that the internal nodes have at least two children, there exists an injective function $f : X \mapsto \mathcal{L}_{\mathcal{T}}$ such that $X = \{(a, c) \mid a, c \in I_{\mathcal{T}}, c \in \mathcal{C}(a)\}$

Proof: [Proof (sketch)] Let a tree $\mathcal{T}$ of depth two, such that all the internal nodes have at least two children. The tree $\mathcal{T}$ is presented on Figure 2.23. The arity of the tree in not important.

Then the set of nodes a of $\mathcal{T}$ (presented on Figure 2.23) such that a is an internal node and a does not have leaf(s) as children $a|a \in I_{\mathcal{T}}, \mathcal{C}_{\mathcal{T}}(a) \notin I_{\mathcal{T}} = \{1\}$. $\forall c \in \mathcal{C}(1)$ such that $c \in I_{\mathcal{T}}$, $\exists$ a function f which associate $f((1, c))$ to a leaf for which c is a child of the ancestor node 1. $\mathcal{T}$ is a tree of depth 2 then each node c of depth 1, has at least two children (considering the assumption on data structure that each internal node has at least two children), or c is a leaf (and c has no children). Thus, f can be defined as follow: $\forall c \in \mathcal{C}(1)$, such that $c \in I_{\mathcal{T}}$, $f((1, c)) = k$ with $k \in \mathcal{L}_{\mathcal{T}}$ and $k \in \mathcal{C}(c)$. For example, on Figure 2.24, $f((1, c_1)) = k_1$.

Then for tree $\mathcal{T}$ of depth two, such that all the internal nodes have at least two children, there exist an injective function $f : X \mapsto \mathcal{L}_{\mathcal{T}}$ such that $X = \{(a, c) \mid a, c \in I_{\mathcal{T}}, c \in \mathcal{C}(a)\}$.

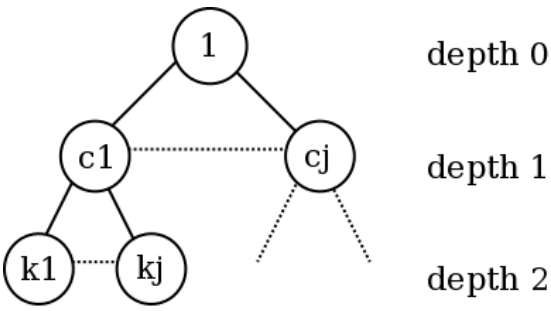

Figure 2.23: Tree of depth two

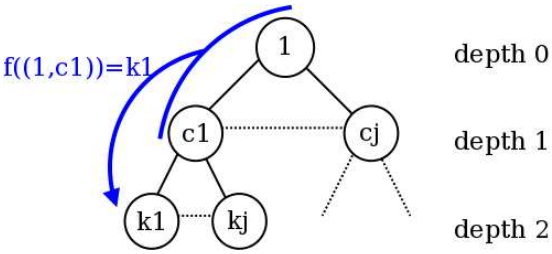

Figure 2.24: Injective function f on a tree of depth two

Assume for a tree $\mathcal{T}_n$ of depth n, there exist an injective function $f : X \mapsto \mathcal{L}_{\mathcal{T}}$ such that $X = \{(a, c) \mid a, c \in I_{\mathcal{T}_n}, c \in \mathcal{C}(a)\}$. We present this kind of tree on Figure 2.25.

Let a tree $\mathcal{T}_{n+1}$ of depth $n + 1$, such that all the internal nodes have at least two children. Every tree of depth $n+1$ can be computed by adding two or more leafs (because in the structure, we assume that all the internal nodes have at least two children) to the nodes at the lowest level of tree $\mathcal{T}_n$ (of depth n).

On Figure 2.25, $\mathcal{T}_{n+1}$ is computed to replace the node k by a tree of depth 1 adding the leaf r and s. It is possible to an injective function define f' from the injective function f definition as follows:
$f' : X' \mapsto \mathcal{T}_{n+1}$ such that $X' = X \cup (r, s) = \{(a, c) \mid a \in I_{\mathcal{T}_{n+1}}, c \notin \mathcal{L}_{\mathcal{T}_{n+1}}, c \in \mathcal{C}_{\mathcal{T}_{n+1}}(a)\}$

1. If $f(x) \neq k$ then $f'(x) = f(x)$,

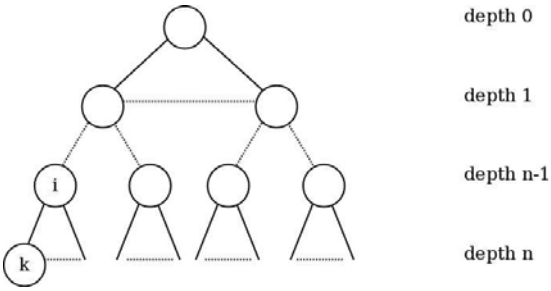

Figure 2.25: Tree of depth n

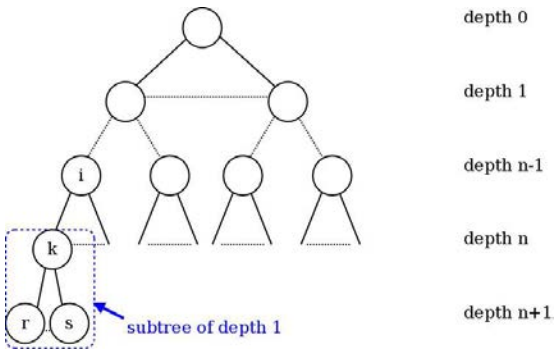

Figure 2.26: Tree of depth $n+1$

2. If $f(x) = k$ then $f'(x) = r$,
3. and $f'(i,k) = s$

In the general case, the function f' can be defined as follow:

$$f' : X' \mapsto \mathcal{T}_{n+1}$$

with:

$$\begin{aligned} X' &= X \cup \{(i,k) \mid k \in \mathcal{L}_{\mathcal{T}_n}, k \in \mathcal{C}_{\mathcal{T}_{n+1}}(i)\} \\ &= \{(a,c) \mid a,c \in I_{\mathcal{T}_{n+1}}, c \in \mathcal{C}_{\mathcal{T}_{n+1}}(a)\} \end{aligned}$$

such that:

1. If $f(x) = k$ such that $k \in \mathcal{L}_{\mathcal{T}_{n+1}}$ then $f'(x) = k$,
2. If $f((j,l)) = r$ such that $r \notin \mathcal{L}_{\mathcal{T}_{n+1}}$, then $f'((j,l)) \in \mathcal{C}_{\mathcal{T}_{n+1}}(r)$,
3. $\forall (i,k)$ such that $k \in \mathcal{L}_{\mathcal{T}_n}$ and $k \in \mathcal{C}_{\mathcal{T}_n}(i)$, then $f'((i,k)) \in \mathcal{C}_{\mathcal{T}_{n+1}}(k)$,

Like $\forall k \in \mathcal{L}_{\mathcal{T}_n}$, $\left|\mathcal{C}_{\mathcal{T}_{n+1}}(k)\right| \geq 2$ (all the internal nodes have at least two children), f' ensuring the following property:

$\forall$ couple of nodes (j,l) such that $l \in \mathcal{L}_{\mathcal{T}_n}$, $l \in \mathcal{C}_{\mathcal{T}_n}(j)$,
and $\forall (i,k)$ such that $f((i,k)) \notin \mathcal{L}_{\mathcal{T}_{n+1}}$
then $f'((j,l)) \neq f'((i,k))$

Then f' is injective.

Then for tree $\mathcal{T}_{n+1}$ of depth $n+1$, such that all the internal nodes have at least two children, there exist an injective function $f : X \mapsto \mathcal{L}_{\mathcal{T}_{n+1}}$ such that $X = \{(a,c) \mid a,c \in I_{\mathcal{T}_{n+1}}, c \in \mathcal{C}_{\mathcal{T}_{n+1}}(a)\}$.

Hence, it is assured that the memory space in the leaves posed as physical objects is enough to hold their hierarchy information.

2.3.3.2 The Application

The leaf nodes holding pieces of information can be used to build the tree structure. Physically, it would represent assembling the complex object using the elementary parts. This is the interesting aspect of our design. To achieve this, information must be distributed and stored in

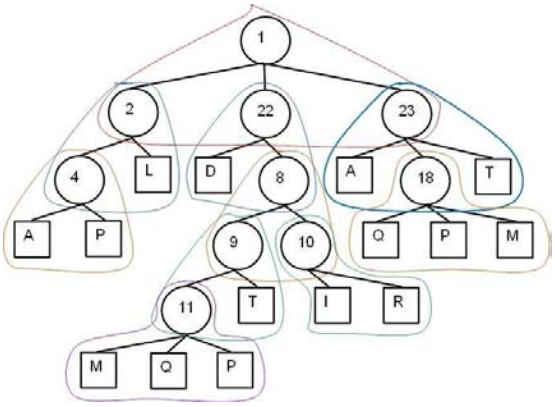

Figure 2.27: Elementary trees marked in coloured circles

the memory of physical objects expressed as leaf nodes. Henceforth, it would be possible to rebuild the tree structure simply using the leaves. The process is elucidated in the later part of this section. The seminal work for our approach is presented in [VBC12] which demonstrates integrity checking of hierarchically coupled physical objects.

The next subsection explains our algorithm to distribute the information while the later explains another algorithm for assembling.

Distributing information among leaf nodes

Now, we would like to introduce the concept of an $\mathrm{Elementary\ Tree}$ (ET) because its group of nodes are correlated. It represents a basic assembly step or process with the nodes involved. It consists of an $\mathrm{internal}$ node with two or more $\mathrm{children}$ nodes. Figure 2.27 shows all the possible $\mathrm{Elementary\ Trees}$ in coloured circles from the example in figure 2.21. Let's pick the $\mathrm{Elementary\ Tree}$ encircled in dark blue from figure 2.27 and postulate some notations for referring the $\mathrm{Elementary\ Tree}$ and its nodes. $ET_{current}$ would mean the current $\mathrm{Elementary\ Tree}$ in discussion. While $ET_{current_root}$ would refer to node with id 23, $ET_{current_child}$ or $ET_{current_children}$ indicates to the descendant(s) of our context i.e. nodes A, T and 18.

This information is stored as to be put into leaf nodes underneath. One can notice that from figure 2.27 that $\mathrm{Elementary\ Trees}$ are connected and the entire tree could be traversed recursively starting from the topmost encircled in red. Every recursive step, an $\mathrm{Elementary\ Tree}$ forms the context with the nodes addressable as $ET_{current_root}$ and $ET_{current_children}$. Algorithm 2 outlines the recursive approach for distributing the tree structure information across the leaf nodes.

Algorithm 2 has a recursive approach to traverse through the hierarchy of ETs. It uses a static stack capable of storing ETs as its basic object at each level. The information associated to an ET consists of the ids of the root and its children. The algorithm starts with the topmost ET of the tree. When the root of an ET is an internal node, the program control moves on to its children. If a child is a leaf, its parent and sibling as well as an ET information (if available) is popped out of the stack as ancestor and written into it. Otherwise, in case the child is an internal

Algorithm 2: Distributing Elementary Tree information among leaf nodes

```
Input(s): 1. the topmost Elementary Tree;
2. an Empty Stack S capable of holding Elementary Trees;
Output: Leaves filled with information to be separated;
disseminateInformation(ET_current, S);
if ET_current_root is NOT LEAF then
    for each child ET_current_child of ET_current do
        if ET_current_child is LEAF then
            store ET_current_root id in ET_current_child as parent;
            store ET_current_children ids in ET_current_child as siblings;
            if S is NOT EMPTY then
                POP(S) for ET_ancestor;
                store ET_ancestor in ET_current_child;
        else
            PUSH(S, ET_current);
            CALL disseminateInformation(ET_current_child, S);
```

node, the current ET information is pushed into the stack and program control traverses using recursion to the make this child node as the root of the current ET.

The recursive algorithm terminates when all the tree nodes are visited in depth-first manner. In the process, the leaves are augmented with information about its own ET (i.e. sibling and parent nodes) and possibly another ET, whose root is its ancestor. Henceforth, the leave nodes contain sufficient information to be dispersed and rebuild the tree structure.

In the next section, we illustrate the procedure to build the original tree progressively using these leaves, irrespective of their order of addition.

Assembling tree using leaf nodes

As discussed above, each leaf node contains two sets of information from which we can build two ETs. The first one would contain ids of the leaf node i.e. itself, along with its siblings and parent whereas the second is an optional containing information about ancestor nodes. The rectangular box in figure 2.28 shows the information sets in form of ETs for the leaf node D in figure 2.27.

Our assembly process consists of two stages described in algorithms 3 and 4 below. We use a list structure L which serves the purpose of temporary cache to store the incomplete parts of tree formed until its completion. The ET containing the ancestor nodes is inserted into list L using algorithm 3 and subsequently the ET of the leaf node.

Algorithm 3 processes the list L and an ET taking both as input. It inspects the root of every part of tree stored across L and tries to move as many of them possible below the ET, wherever the nodes match. The modified list L and ET are then passed on to algorithm 4 for further processing. In this phase, the algorithm strives to insert the ET into any of the subtrees

Algorithm 3: Adding partially formed tree(s) in list L below an Elementary Tree from leaf node

```
Input(s): 1. an Elementary Tree;
2. a List L capable of holding Elementary Trees;
Output: arranged Elementary Tree and List L;
arrange(ET_current, L);
for each child ET_current_child of ET_current do
    for each i from 0 to L_size-1 do
        if ET_L[i]_root id equals ET_current_root id then
            Move ET_L[i]_root to ET_current_child;
```

across list L. If it is unsuccessful, the ET is augmented to L.

Figure 2.28 depicts the ET with ancestor nodes of leaf D being augmented to L by algorithm 4. The curved dotted arrow in figure 2.29 shows node 8 from the list L being moved under the ET using algorithm 3. This modified ET is inserted into the list L by algorithm 4, shown as straight dotted arrow in the figure.

Algorithm 4: Inserting an Elementary Tree from leaf node into the list L

```
Input(s): 1. an Elementary Tree;
2. a List L capable of holding Elementary Trees;
Output: List L with an Elementary Tree inserted;
insert(ET_current, L);
set found to FALSE;
for each i from 0 to L_size-1 do
    if ET_current_root found in tree at L[i] then
        set found to TRUE;
if found is TRUE then
    Insert ET_current into tree at L[i];
else
    Add ET_current at L_size;
```

Termination criteria to detect tree completion We have mentioned that throughout the process, we use a list structure where the leaves are added incrementally to progress towards assembling the complete tree. But, there's one question that would come to the readers mind is, figuring out the completion of the tree structure. When we have a leaf node to be added, we have used markers for nodes in the ETs constructed to differentiate between an internal nodes and a leaf. In figure 2.30, for example, the ET in the first cell of the list originates due to leaf node M, which is marked differently. When its sibling leaf node P is added using algorithm 4, it performs a merge operation to form the structure encircled in figure 2.30. This

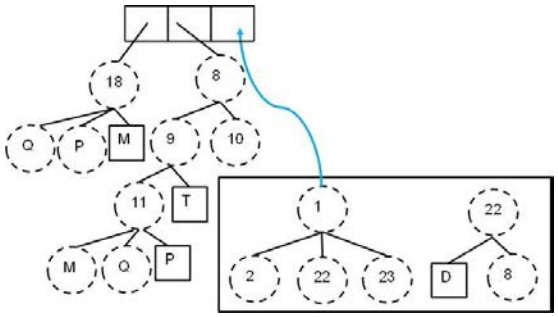

Figure 2.28: Elementary trees added to end of List L

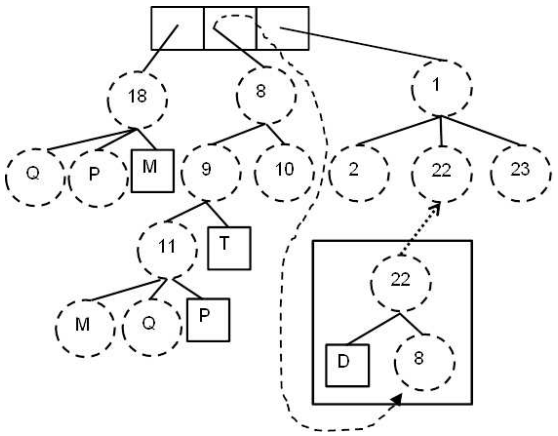

Figure 2.29: Tree from List L added under Elementary tree and then inserted into List

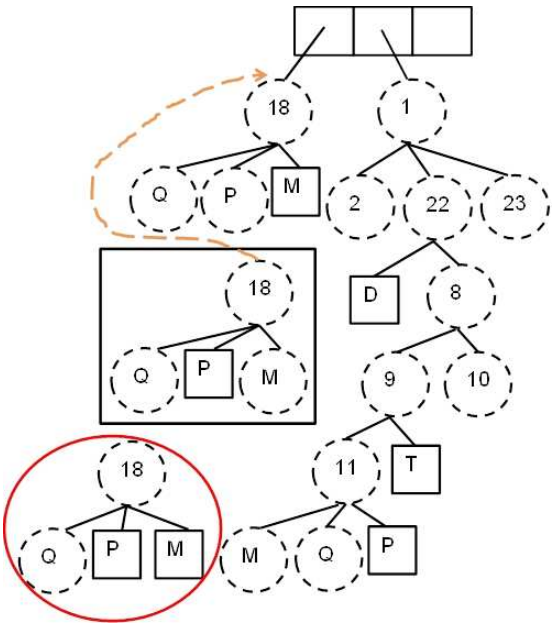

Figure 2.30: Adding sibling leaf nodes

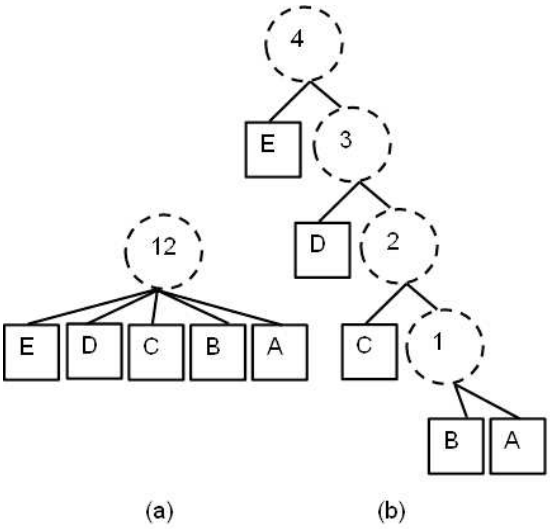

Figure 2.31: Comparing structural composition of nodes

differentiation of the nodes help us to infer if the tree structure pointed to by a cell of the list is partial. Additionally, if the list contains only one tree structure and all the leaves are marked, then it can be concluded to be the complete tree.

Evaluation

Our approach for storing the description of tree hierarchy into the leaves. This could be used for applications where there are real physical entities forming a hierarchy for some purpose. These entities obviously correlates to the leaf nodes of our tree structure whereas their internal nodes would be virtually existing. Hence the leaves plays a very important role in the structure. Since they are physical entities, they are also vulnerable to be damaged, lost or corrupted. Hence providing resilience is very important to safeguard against these circumstances. Reiterating our assumption, that an internal node must have at least two children and these siblings store information about each other. Hence, if at least one among the children endures the loss, the missing ones could be identified. Apart from providing resilience, our method provides support for checking the integrity of structure as well.

We have performed simulation of our approach with random trees of various size and order. Since we have assumed an approach where internal nodes have unique ids, our tree generator requires to verify this constraint. Trees up to 15 level with 5 children could be easily generated having around 150 leaves (permitting duplicates) but fulfilling the restriction regarding unique internal nodes. They could be easily distributed with their hierarchical information into the leaves and assembled successfully. They have performed as we had desired.

Every node in the tree structure described here, are represented by identifiers which are stored in the leaves to express their relations. Hence, leaves represented as physical entities would have a limitation on the memory space to store the information. So the length of the identifiers should be chosen carefully depending on the application area, memory capacity of the physical entities and an estimate of the tree size. The leaves in our approach have provisions to accommodate two sets of data i.e. ETs. Let's consider two different ways of building the tree structure as shown in figure 2.31. Both diagrams represent combining leaf nodes A, B, C, D and E. While diagram (a) depicts all of them combined as siblings in one level, diagram (b) performs the same at multiple levels. So, the first tree structure would have space unused, reserved for ancestor contrary to the second case. The multi-level tree structure would have the leaves containing heterogeneous information relating to one sibling, parent and ancestor nodes. Hence, depending on the pros and cons, we should make an informed choice if we want an assembly of physical entities all in one or multi-levels, depending on our application area as well.

To compare these two situations for introspection and verify the impact on the memory usage, we performed a different set of simulations. Random complete trees were generated using java with depth and children up to 10. To give an estimate about the size of complete trees, typically one with depth 6 and 5 children at each level would have 15625 leaves at the lowest level. When the depth is increased with the number of children constant, the tree leaves grow exponentially than the other way round which is multiplicative. We calculated the average number of IDs stored per tree leaves and made two contrasting observations. We can observe in figure 2.32(a) that the rate of increase for this average is steady, if the depth is increased

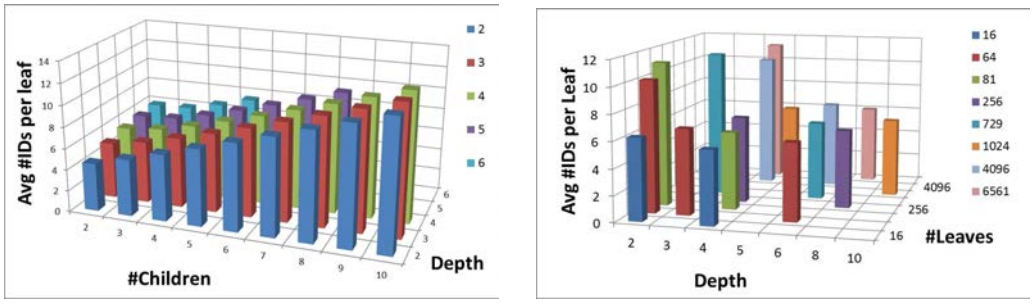

Figure 2.32: (a) Memory usage of leaves in complete trees; (b) Increasing depth consumes less memory in leaves.

keeping the number of children constant. In figure 2.32(b), we have highlighted that when the depth increases, the average value decreases for the same number of leaves.

2.4 Conclusion

In this chapter, we explained the core contribution of this thesis. The principle approach is that objects self-support information systems that enable them to participate in pervasive applications locally. The first contribution demonstrates how objects could be self-described using “Tangible Data”. The second contribution illustrates how group of related objects are digitally linked using data structures, with their information distributed over the objects. Properties involving the group can be locally validated using the structure and their physical relation justifies the name “Tangible Data Structures”.

Chapter 3

Tangible Data in Waste Management System

In the last chapter, we explained our contributions of self describing physical objects that enable them to participate in local interactions. We have discussed how RFID technology suits to our approach due to its properties and low cost. Thus everyday used objects can be made smarter easily for them to interact in smart environment.

Following our approach, we have demonstrated some of its utilities in the context of waste management system. The exploratory project Bin That Thinks assumes the existence of smart waste i.e. the self described objects when discarded as waste. Thus the necessary information required for handling these items would be available locally, at every stage of the waste management system. Machines can make decisions on site based on the tangible data available pervasively from these items. They can operate autonomously with high scalability. As the waste items are collected in containers initially, they are screened and channelized for proper sorting. Additionally, information from collection of smart waste is aggregated and stored as tangible data on the trash bags. It serves as self description of the bag with collected item and propagates the information forward in a manageable way for the later stages in the waste processing system. In this short chapter, we have illustrated how our approach could be leveraged in the various stages of waste management system. We have described the flow of waste along with the transformation of its tangible data. Before getting down to the nitty-gritty details of our smart waste management approach, we have provided a section that briefs about the project.

3.1 About Bin That Thinks [1] Project

This project was financed by ANR (l'Agence Nationale pour la Recherche) during their call ECOTECH 2010. Its tenure was for three years starting from December 2010 to be executed in phases. The work was collaborated among various stakeholders as below:

1. http://binthatthink.inria.fr/

- Two teams from INRIA [2]: ACES [3] and POPS [4] situated in Rennes and Lille respectively.
- Véolia Propreté [5], which is a major waste management organization in France. They have contributed with the waste domain knowledge for better understanding of the feasibility of solutions.
- Etinéo [6], a startup company focused on M2M communications and ambient networking located in Lille.

Waste management is an arduous task given its growing amount and their variety of approaches depending on the type of waste. Waste prevention, recycling/reuse and disposal are the three principles laid down by the European Union's approach to waste management. While the first principle avoids the generation of waste, the remaining intends to take care of the waste produced after products are used. There are dozens of materials that are recycled for better sustainability. Products made of aluminum, plastic, cardboards, steel, glass etc. can mostly be recycled. Since these materials could be extracted out of them, the irony is, they are valuable in spite of being waste. Commodities like batteries, electrical and electronic wastes, packaging waste, used oil cans require special attention for recycling and reuse. Some of these items can be hazardous to be dropped into a waste bin containing other items for disposal. So prevention of such instances would avoid physical hazards as well as degrading the quality of already collected items.

Sorting is a process in waste management, performed for efficient recycling and reuse of the waste materials. Different waste items contain recyclable and reusable materials having economic value. Hence, sorting them at the earliest has advantages. Moreover, it also ensures the quality of the collected waste with no contamination. It also avoids potential hazards. Transporting them to the sorting facilities could be another overhead to cost if not performed early. Lastly they might end up as landfills, if left undetected.

"Bin That Thinks" assumes that all products would be RFID tagged which is partially visionary. The tags contain self-description of the product. Currently, manufacturers and retailers already tag many of their items for tracking. We have Smart Bins equipped with RFID readers and embedded processors. Some of them are optionally equipped with sensors and wireless communication. The bin can query using RFID and have an overview of its contents. The processor can compute and decide on various aspects using this information. Its RFID reader can sense an object brought near it and read the data on its tag. The bin actuates by opening its automatic lid, if the item is inferred suitable. Each Smart Bin communicates with its neighbours using short range protocols. Every bin acts like a node in multi-hop wireless sensor network (WSN). They exchange information on their contents, detection of hazards etc. to notify operator's nearest base station or waste collection vehicles passing-by. While many are small household bins, few among these are larger community bins. These are equipped with sensors to detect physical hazards, GPS to locate their position and GPRS ability to communicate directly with the operator seeking immediate attention during emergency situations. Figure 3.1 shows the similarity between the Smart Bin infrastructure and a WSN. While most

2. http://www.inria.fr/
3. http://www.inria.fr/en/teams/aces
4. http://www.inria.fr/en/teams/fun
5. http://www.veolia-proprete.com/
6. http://www.etineo.fr/

of the communication among the entities - like smart bins and collection vehicles - take place using short range multi-hop technology, long range GPRS communication is equipped in few entities of the infrastructure. The latter is only used between operator's remote station for reporting emergencies and monitoring the status of areas.

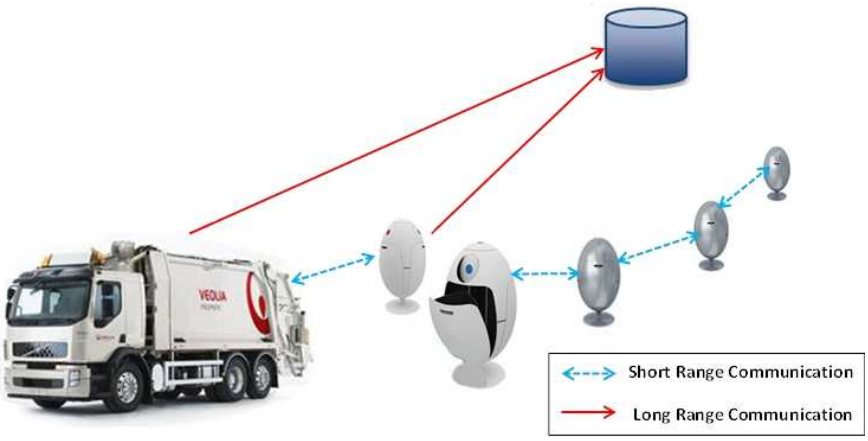

Figure 3.1: Smart Bin Infrastructure as a Sensor Network

The project's approach has an unique way of sorting that confirms to the guiding principles of waste management as well as provide cost benefits for business enterprises. It is designed to work in such a way that the sorting process is enforced the moment items are trashed. Eventually, wastes are sorted in stages so that they are channelized appropriately while traveling downstream in the waste management chain and reach their proper destination. This scheme reduces the load on sorting centers, associated costs and risks for operators. It has the following benefits:

- Wastes are identified and refused at points of collection due to inappropriateness; this ensures that quality is maintained.
- They are sorted implicitly in stages increasing the value with less overhead; hence benefiting the operators from business perspective.
- operators would know the types of waste present in the bins (over a large area) at real time due to the infrastructure; so the waste collection vehicles can optimize their route while making rounds. Different waste materials have economic value that are collected and sold by the operators. It also enables them to monitor potential hazards and attend to it in due time.
- In the process, users are educated with proper handling of waste and grows the habit of dropping them in the correct bin.

The responsibility of research and development of the project had been shared among the two teams - ACES and FUN - based on the areas of research interest. The focus of our team was primarily on the working of an individual Smart Bin while the other was responsible for energy efficient M2M communications. Messages are transmitted among the smart bins mostly for informing the operator and monitoring purposes. Apart from getting occasional alerts, the

operator would be able to have a global overview of the situation across the city at any point of time. They can be aware of the bins filled up in an area or have statistics of the contents collected. An operator's collection vehicle can optimize its route with maximum waste collection. While accepting waste from the users, the smart containers have the ability to function and process information autonomously without absolute communication. They are designed to capture pieces of information available pervasively and combine them into meaningful data for making inferences and use by the embedded applications. The pervasive information is available from the data stored in the RFIDs, tagged to the participating objects. This thesis - funded by the "Bin That Thinks" project - contributes with techniques to use RFIDs as small storage spaces providing information pervasively for autonomous processing. Having autonomy is one of the desirable properties but there are various challenges for achieving them. The limited memory capacity of the tags is one of them.

3.2 Waste flow and global architecture of the waste management system

In the last chapter, we had elucidated our approach of self-describing objects in the physical world. Most of these everyday used objects undergo through various phases of its life cycle; from its manufacturing until disposal. Presently manufacturers and retailers already use RFID tagging of their products extensively for inventory. We have assumed their use would be extended with self-description to make them smarter. This would enable them to participate in smart interactions during their product phase as well after their disposal; once they become waste. Hence the tangible data for these items would be available pervasively for autonomous processing throughout the waste management chain. However, it might be necessary to have some exclusive information for products that would be useful for their proper disposal as waste.

The waste management architecture we consider is built around several elements: waste items, domestic bin, trash bags, collective containers and collecting vehicles. The waste flow starts from the waste items and the domestic bin to end in the collecting vehicles. We now describe each of step in the waste flow and how elements interact.

3.2.1 Wastes description

The system described here is based on a self-describing approach of each waste. We propose to associate digital information to each waste to ensure an appropriate treatment of each item. This is a key point of the proposed approach.

In the selective sorting process, the type of a waste item is identified by its principle composition. For example, a plastic bottle is identified as a plastic waste, and a cardboard box is identified as a cardboard waste. In the presented approach, each self-describing waste carries digital information about its type. Other properties of the waste are interesting during their collection process. For example, the weight of each waste can be used to estimate if a bin is full, or empty. Without using measurement sensors, the weight data of a waste item can be stored in digital information attached to it, making itself describing.

3.2.2 Wastes identification

The user is the primary actor in the selective sorting process. Based on this observation, our waste management system proposes some pervasive assistance for the selective sorting process. Then, the waste flow presented in Figure 3.2, begins at the user level where the trash is generated. As shown on the top of Figure 3.2, we propose to favour a behavior of the users: by indicating the appropriate bin for a waste, or more directly, by opening the lid of the bin corresponding to the type of the waste.

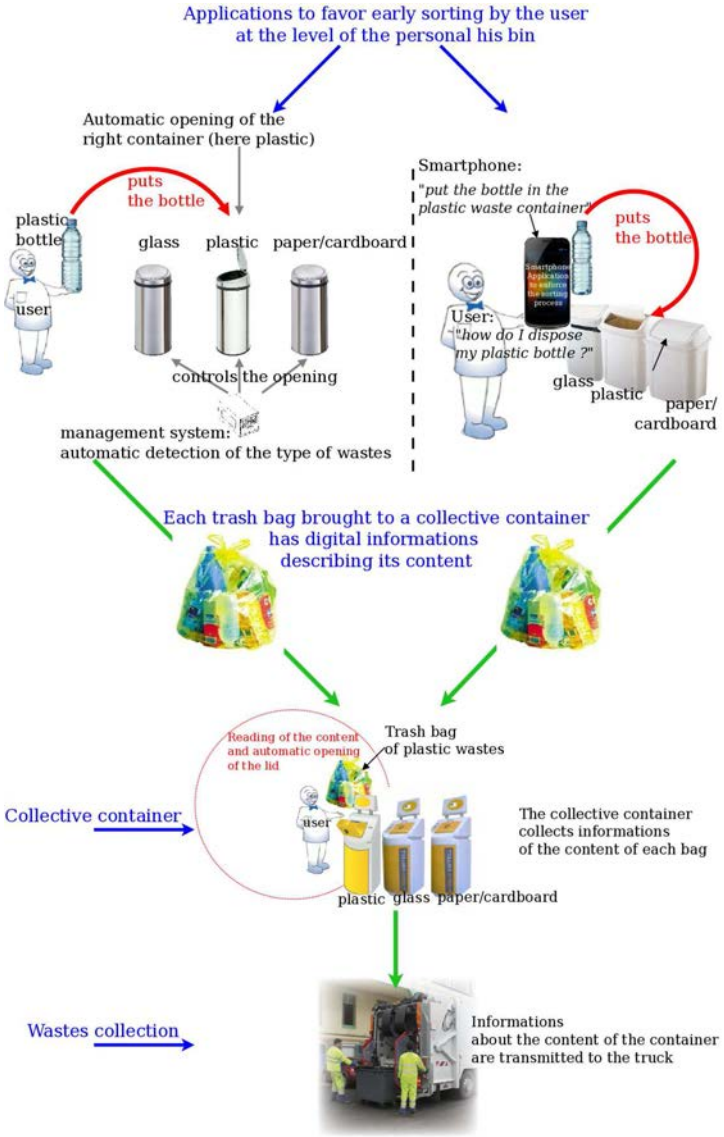

Figure 3.2: Waste flow and global architecture of the system [GC13]

3.2.2.1 Individual smart bins

At the first step of our waste sorting system, the information contained in the RFID tag associated to each smart waste is used to help the people discarding an object in the appropriate container. Here, the main goal is to reduce the sorting errors when someone does not know which is the right container, or mistakenly discards the object in the wrong one. It is also help people to learn the selective sorting rules applied locally. The smart bin system uses the self-describing approach of smart wastes to improve the selective sorting quality.

The description of smart wastes is stored in a RFID tag physically associated to each smart waste. Using a RFID reader, the smart bin reads the RFID tag attached to each smart waste to determine the appropriate treatment. Let's consider the example of someone who wants to discard a plastic bottle in a bin. He puts the bottle near a smart bin as it is shown in top-left part of figure 3.2. When the plastic bottle is in the antenna area, the tag associated to the bottle is detected. The data stored in the tag is read to determine the appropriate procedure to discard the bottle. If the bin accepts plastic objects, then the system opens the lid of the bin. Otherwise, the system keeps closed the lid of the bin.

Note that it is also possible to control the opening of several containers using a single RFID reader. The top-left part of figure 3.2 presents a prototype of a selective bin. In this approach, a management system connected to a RFID reader uses the data stored on waste tags to open the correct container. In this example, when someone want to discard a plastic bottle, the container for the plastic wastes is opened by the management system. Among the three containers, the one accepting plastic items would open and all the remaining will remain closed.

This approach assumes that the management system tracks the information of the wastes that are discarded in each container of a selective bin. When a waste is discarded in the container, the management system updates the memory inventory for this type of waste. In this way, undesirable wastes for a given container are either rejected or tracked, depending on the chosen policy for handling undesirable wastes. The scanning of products is done item-wise to ensure a complete reliable reading process.

Figure 3.3 presents a prototype of a smart bin approach based on UHF RFID technology, which is already used in the supply chain (inventory) management systems. As per our futuristic assumption discussed before, the UHF tag is placed on the packaging (by manufacturers and retailers) of each product at the beginning of its life cycle. Since such a tag is already attached on the packaging of each product for the supply chain management process, we are interested to reuse the tag and its technology in our smart bin approach.

3.2.3 Cheaper alternative using QR codes

Our self-describing approach presented using UHF RFID tags is very futuristic. It would require the ubiquitous tagging of products with RFID which is a matter of business interests for organizations over time. Although a tag costs as low as $0.25, the business entities must perceive enough monetary benefits to recover the cost of affixing tags for trivial objects. Those benefiting might even agree to share on the cost. The deployment of collective bin presented in section 3.2.2.1 could be a costly affair initially. It is a matter of time to bring down the costs due to cheaper hardware and mass production. Given all these factors, in this section, we introduce

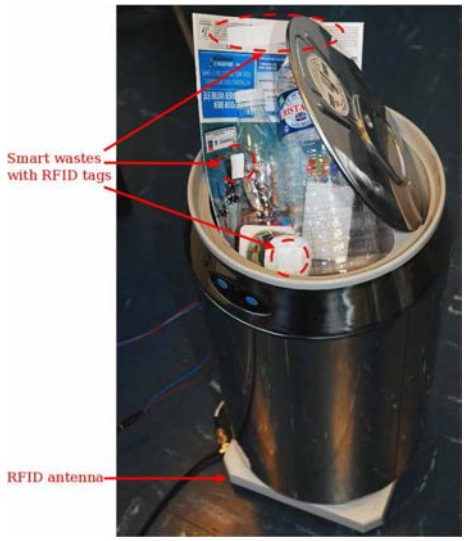

Figure 3.3: A collective bin using the RFID technology.

a cheaper solution using QR code technology that would allow an early adoption of some of the concepts and applications presented previously. It serves as a quick alternative approach to self-describe objects at almost zero costs. The approach can also be used in complement to RFID. However, this approach requires the NFC sensor embedded in the smartphone of the users.

This alternative approach assumes that every waste is associated to a QR code describing its type. The mobile application maintains in its memory the current inventory for each type of collected wastes (for example, 3 inventories if there are 3 types of collected waste). Waste disposal would require users to scan each item, allowing the mobile application to update the current inventory for this type of waste in phone's memory. Some other waste properties, such as weight, could also be collected at this step.

A smartphone is a small, low-cost, mobile computer. Moreover, most smartphones now embed a camera enabling them to read bar codes or 2-dimensional QR codes (also known as "flash codes"). A first step in the solution would consist to scan a QR code (or bar code) associated to a product, and to use this information for giving a sorting instruction to the smartphone of the user. As in the approach of the individual RFID bin presented in section 3.2.2.1, it is also important to report the actions of the user to the waste collecting chain.

In the top-right part of figure 3.2, a user wants to dispose a plastic bottle. He scans the QR code associated to bottle. The properties associated to the bottle are added to the inventory of the plastic container that is stored in the smartphone's memory.

Obviously, reading QR code is less convenient than RFID reading. In addition, in this approach the opening of the lid is also not controlled by an automatic system. However, this approach allows the deployment of the rest of the chain without requiring the smart bins presented in section 3.2.2.1 inside each home. Beside being cheaper, the mobile application also provides helpful support to the user regarding the selective sorting rules in application.

Like the individual bin presented in section 3.2.2.1, the management system of the collec-

tive bin tracks wastes properties as they are disposed. When a smart trash bag associated to a RFID tag is dropped in the collective container, the management system updates the collective inventory according to the new bag's content. Prevention of sorting errors is also possible, provided that the user actually fills his trash bags according to what he scans.

We don't rely a network connection on the bin instead it is the waste bag itself which will store the waste inventory. As we will see in next section.

3.2.4 Trash bag

To ensure an appropriate treatment, the knowledge of the type of wastes contained in a trash bag is crucial. As for the wastes, it is also possible to associate several properties of each trash bag: for example, the owner of the trash bag, and the number of items in the trash bag can also be considered. In the least, some digital information about the total weight of the trash bag, its content and the number of items contained in the trash bag must be physically associated to each trash bag. Digital information associating the identifying the owner of a trash bag would be interesting for statistical data of consumer waste production. The tangible data of a trash bag is defined by an analysis report of its contained items.

The analysis report stores some important information for the selective sorting process. The information stored in the analysis report is to determine whether the trash bag could be accepted. In Figure 3.2, this analysis report is transmitted to the collective container, when a user brings a new trash bag.

3.2.5 Collective container

In our waste management system, each collective container is associated to an embedded computing system which processes the data of the analysis report of each trash bag, making it a smart bin. When a new trash bag is added in a collective container, the analysis report is read.

Considering the type of wastes contained in a trash bag, a collective container determines whether it could accept a trash bag or not. It verifies for its suitability with regards to the already present items in the container; for contamination or physical hazards. For example, a collective container collecting only plastic wastes can stay closed when a user brings a trash bag containing the cardboard objects: it would only open for a bag of plastic wastes. If the trash bag is accepted, the smart bin stores some information about the content and owner of each trash bag. Then, the content of a collective container is iteratively updated as a new trash bag is added. The information stored by the collective container is transmitted to the truck during the collection by using a local connection, as it is presented on the bottom of Figure 3.2. At this step, the errors of the selective process can get transmitted. Among the collection of wastes, the highly polluting wastes which are not placed in the appropriate container, are detected: for example, it becomes possible to detect a battery placed in the container dedicated to plastic waste.

The waste flow takes place through the described stages. They are sorted and separated at every level. Hence it avoids the mammoth task of sorting everything in the later stages. Thus reducing on the transportation overhead cost to the processing plants, as well as reduces on their process loads.

3.3 Prototype demonstration

Figure 3.4 shows the prototype developed out of the collaborative project "Bin That Thinks" [TA]. It consists of prototypes developed for the various solutions of the waste flow architecture discussed in this chapter and are marked numerically. The first shows two kitchen bins that collecting different types of waste. The QR codes are scanned using the smartphone placed in-between which is also used to write the inventory onto a NFC tag when a trash bag is sealed. The second is the collective container that can scan the inventory tag and verify with the bag contents for compliance. Its optional screen displays the status and statistics. The later two shows the required applications deployed for monitoring in the truck and operator's station respectively. While the former visualizes information about bins to be collected, the later displays the waste management infrastructure over the city map. Appropriate operations could also be done through these applications. Finally, the entire demonstration video can be viewed at [bin].

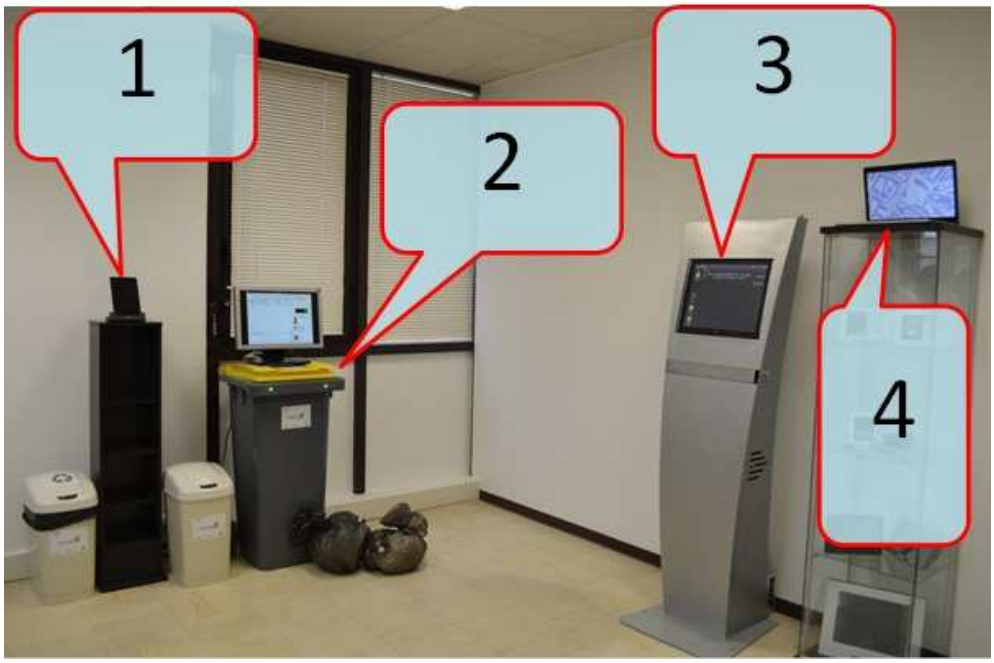

Figure 3.4: Prototype demonstration: (1) Kitchen bin (2) Collective bin showing status (3) Truck's Application Screen (4) Operator's status application at city level

Chapter 4

Conclusion

Pervasive computing is a trend post the desktop computing most prevalent previously, where the real world was manifested using Graphical User Interface. The interface between the physical and digital worlds had a distinct separation. However with the infusion of the new trend where these two worlds are integrated, their distinction would eventually fade out. The physical world environment is embedded with digital information and processing. However, there are concerns that need to be addressed. The most common approaches use network connected to centralized infrastructures.

In this thesis, we have demonstrated an approach contrary to the commonly used. The data is distributed in the physical environment. The objects are piggybacked with the information using RFID tags which enables them to take part in the local interaction. One of its important advantages over the centralized is that it does not require network infrastructure. The information stored centrally which requires to be accessed using the infrastructure. Such centralized architectures require to maintain better security and privacy. We have illustrated two different ways in which the approach could empower the objects to interact locally.

In the first instance, we have explained self describing objects with tangible data in their tags. It would make everyday used objects smarter to participate in local pervasive interactions. The data describing a physical object could be varied from its status, properties of relevant qualitative/quantitative description useful throughout its life cycle; from its manufacture until recycling. The challenging issue is the limited memory space of RFID tags cannot hold unlimited amount of tangible data. So the relevant data must be encoded for the objects to participate in selective applications. We have demonstrated the benefits using some encoding schemes using hexadecimal or numeric characters. They are customized depending on the data to be described which would be supported by selective applications. Self describing the objects using standard domain knowledge and reusing them throughout the object's lifecycle is another way to reduce redundant data. Thus another way to address the challenge of memory limitation while increasing interoperability. This gives an advantage of autonomy to the object to participate in a variety of pervasive interactions. Secondly, security and privacy is ensured as the information is tightly coupled with the physical objects. Traditionally, RFID tags store unique identifiers that point to relevant data in the cloud which could have a potential breach of security and privacy. It could allow generating access patterns and profiling by the servers

or third parties. Cost is also among the important advantages as the proposed approach that enables a retrofit to make objects smarter by affixing tags that cost few cents; whereas the traditional ways require costly setup and network infrastructure. Our approach can be put to use in the near future given that there is an increasing trend of tagging objects with RFID and privacy is a growing concern among users.

The second instance utilizes the principle of tangible data that includes multiple objects. We have discussed that objects are complex or related in most cases. If all these individuals are tagged, we can leverage the situation by optimizing the distribution of data across their tags. The other important feature is the association of these individual objects in the physical world. We illustrated how this association could be embodied by the link created among the tangible data distributed across tags. It could be naturally represented using forms of data structure as linked list, graph, tree etc. We have called them tangible data structure. It opens up for various kinds of pervasive services that keep the coupled objects together; like verification of missing objects, assistive services etc. The concern for security and privacy still remains addressed, since the tangible data structure point to other related objects. The contributions argued in this thesis have been amply illustrated with examples wherever necessary. Additionally, we have demonstrated their utility in two application domains; self-description for efficient waste management and tangible data structure for assembling complex objects.

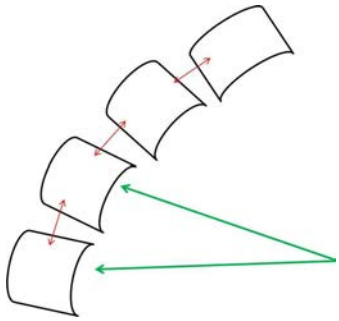

Figure 4.1: Sectorwise inventorying of Smart Library using beam-forming RFID antenna

4.1 Perspectives

This thesis has defended an approach of embedding tangible data along with the objects to make them smarter. RFID technology is still not matured enough; thus has many limitations. However our approach opens up a new direction of making objects smarter while not compromising on the security features. The semantic digital data remains pervasive and ubiquitous within the physical environment for access. Due to the tightly bound nature of the approach, the higher level context and situation could be perceived and would be altered dynamically with its physical changes. Physical interactions analogous to the digital operations like searching, sorting etc. could be performed within the environment. Thingful is a service launched by a start-up recently that hopes to perform activities like Google® on Internet of Things [Thi]. Apart from the tag memory limitation, RFID antennas needs improved reliability to address the

physical objects within a physical space. Currently, the power and direction of RFID antennas can be adjusted. Their focus of radiation could be from highly directional beam to omnidirectional. Intelligent controllers could be developed to exploit these properties leading to smarter infrastructure; to sense the local space and actuate. Consider a prospective example of a library as a smart space that can read the tangible data within the environment. It is equipped with interrogators that can perceive individual physical objects and offer smart services; search for books, verify if they arranged properly etc. Speedway xArray is a RFID reader system recently developed which can monitor a wide-area physical space through its beam-forming antenna array [RFI]. Figure C.7 shows the mechanism how individual physical objects - in this case, books at smart library - could be addressed within a smart space using such a RFID reader. The green arrow shows the directed nature of the RFID antenna. The entire physical space could be partitioned in sectors and objects addressed on an individual basis for various applications. It can address physical objects individually, shown by the blocks in the diagram. The access is enabled to traverse related objects with the help of tangible data structure.

Another perspective to this thesis is designing a tool for building Tangible Data Structure. The idea is providing a recommender system that would aid a novice user to build their customized structure. It would consider design parameters like available memory of different tags, importance of each physical part, targeted application to be provided etc.

Finally, as discussed previously, the approach in this thesis aims to do pervasive computing locally. We advocate due to their advantages of scalability, privacy, security etc. However, in the process we do not disagree with the potential of pervasive computing using external infrastructure. Collaborating the two approaches would enable a robust pervasive system. It could be complementary providing uninterrupted minimal pervasive services in the absence of external infrastructure. Alternatively, the collaboration could be hybrid, similar to IBM® edge computing.

Appendices

Appendix A

Smart Bin using OWL Ontology

A.1 Detailed System Design

A.1.1 Input

New waste items are added to the system. They are affixed with RFID tags only for the purpose of identification by the system, which contains a RFID reader for scanning. The tags do not contain any such data that has privacy concerns. Mostly they contain the category information.

A.1.2 Knowledge Base (KB)

Machines can be made to perform reasoning effectively provided it has the necessary knowledge, which is machine readable. In cases of large domain knowledge with lots of factors influencing the reasoning, using machines should have extra benefits. Using ontologies are a very good way to serve the purpose [MESG10]. An ontology consists of common set of vocabulary as shared information of a domain. It includes machine-interpretable definitions of basic concepts in the domain and relations among them [G+93]. Lately, the development of ontologies has begun to find many uses outside the Artificial-Intelligence laboratories. They are being commonly used on the World-Wide Web and finds applications for sharing information widely in the field of medicine.

The Web Ontology Language (OWL) is a World Wide Web Consortium (W3C) Recommendation for representing ontologies on the Semantic Web [(W3a]. Presently, there are a lot of ontology editors for OWL. Among them Protégé is a Java based Open Source ontology editor. We used Protégé since we found it to be an efficient and user-friendly tool to prototype our ontology rapidly. During the ontology development phase we visualized the graphical representation of our OWL ontology on the editor. The comprehensive Java API provided by Protégé [KHM+05] was also an added advantage while developing our stand-alone application in the later phase.

We have used an ontology based approach for the KB for the reasons stated above. The properties causing incompatibilities must be described in the ontology. Apart from these, other

information like conditions in which the categories are incompatible, possible hazards of incompatibility etc are also stored in the ontology.

Due to the advantage for describing a domain easily, we have used ontology based approach for describing the waste domain. The ontology contains description of various categories with the conditions for hazardous properties. This constitutes as the initial knowledgebase of the system, which updates itself as new items are added.

For detailed demonstration, we have used a sample OWL ontology using few hazardous properties, conditions including an external condition to demonstrate the inference of incompatibilities between objects.

We start with building our KB using Object properties of OWL ontology that would represent the conditions. As mentioned earlier, a category is described to have various properties under three different conditions. Mapping and comparing Figures 2.8 and A.1 would make the idea very clear.

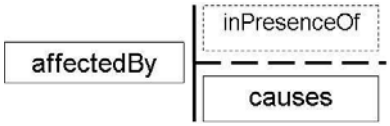

Figure A.1: Object properties as mapped in OWL

Next we define two Data properties in the ontology namely hasStatus and hasTemperature. While the first one can store values of type boolean and acts as a flag, the other is used to hold integer data as in Figure A.2. They are used to express the external conditions as explained subsequently.

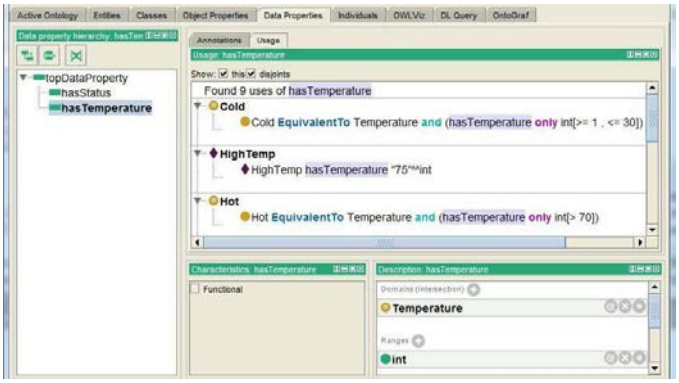

Figure A.2: Data property *hasStatus*

The left side of Figure A.3 shows the representation of a domain knowledge using hierarchy of classes in ontology. The class Waste is the topmost level in our ontology structure and hence, all its subclasses would represent information of this domain. Following are the description of its subclasses and the information it holds:

– Properties list the various instances representing the hazardous properties of the domain. It also includes some external ones like temperature, pressure etc.

– Categories represent the classifications of the waste domain. Each of these categories are represented as classes in ontology along with the description of conditions. The RFID tagged waste items added contains reference to these categories. They are added to the system as individuals of the referred subclasses.
– Hazardous contain all the incompatible or hazardous items. It's subclass selfHazardous holds items that could pose hazardous by itself. They would be subsumed by the upper class. The user can glance through all the incompatible objects added to the ontology.

The right side of Figure A.3 displays subclass Properties containing the list of seven properties that are possible with this domain. Examples of OWL individuals such as Explosion, Flame etc. are some possible properties. None represents a special kind of property, which indicates no conditions at all.

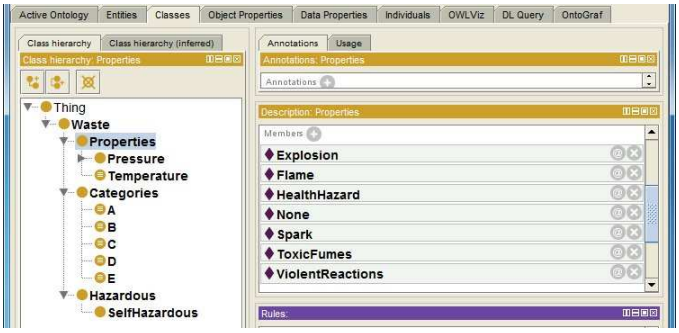

Figure A.3: Sample Ontology with Classes and Conditions

The Figure A.4, we have listed examples of five different types of categories named A, B, C, D, E subclassed under Categories as examples. It should be noted that each of these five classes represent the types of categories possible in the domain described with the conditions of hazardous properties. They are described in form of relationships with individuals of the class conditions using the OWL object properties.

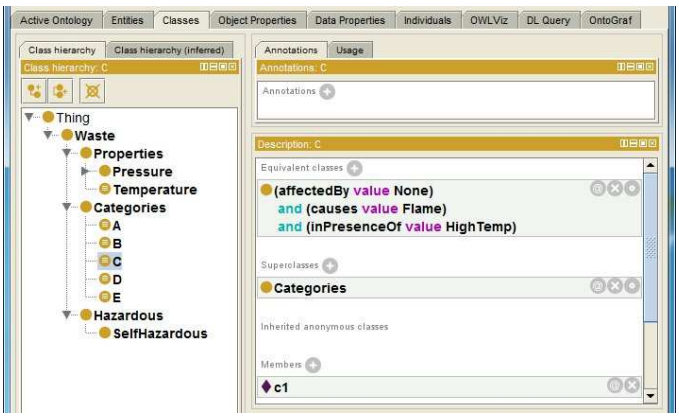

Figure A.4: Describing Categories

In Figure A.5, shows an example to represent temperature as an external condition for in-

compatibility of objects. In this example, we have three individuals of the class Temperature as LowTemp, ModerateTemp and HighTemp. They are linked to the two data properties, hasStatus and hasTemperature. The hasTemperature property basically defines the temperature ranges it represents. The hasStatus property can be set to true for any one of the instances which would indicate that external temperature around the system. Hence, from this way of representation we can indicate discretized levels of some external conditions and the one prevailing around the system.

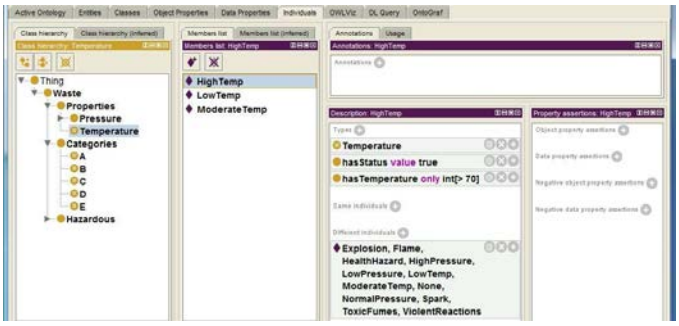

Figure A.5: External Conditions

The class Hazardous consists of a subclass SelfHazardous. SelfHazardous contains all the items that are self hazardous which would be subsumed by the upper class as well. Additionally the class Hazardous holds all pairs of incompatible items that are inferred hazardous. In fig A.6, we see there are currently one pair of item c1, d1 that exhibit possible incompatibility under an external condition of HighTemp.

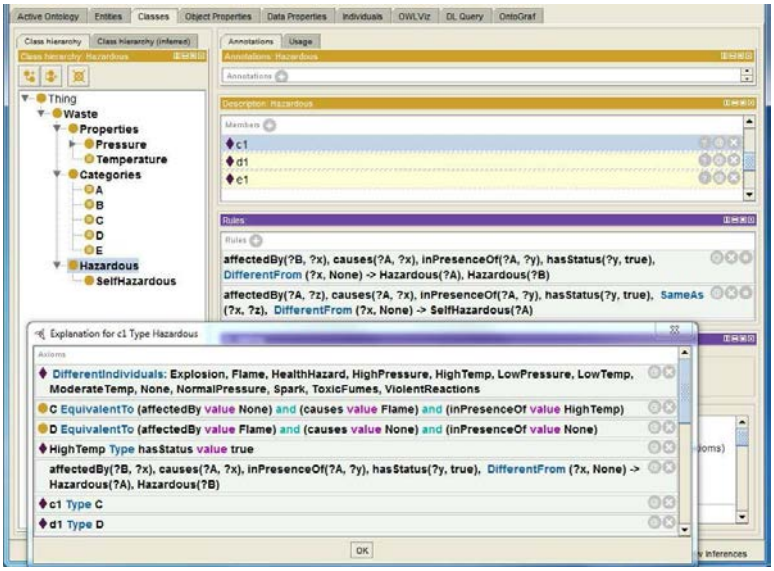

Figure A.6: Dashboard

A.1.3 Reasoning/Rules

Given the ontology, which acts as a KB in our architecture, we are all set with having all the necessary information at hand to reason out something useful i.e. the objective of inferring incompatibility or hazards. In the recent years, rule languages have been added on as a layer combined with ontology in order to enhance the reasoning capabilities. Semantic web Rule Language (SWRL) is used to write rules expressed in terms of OWL concepts and for reasoning about OWL individuals. It provides a deductive reasoning specification that can be used for inferring new knowledge from the Knowledge base.

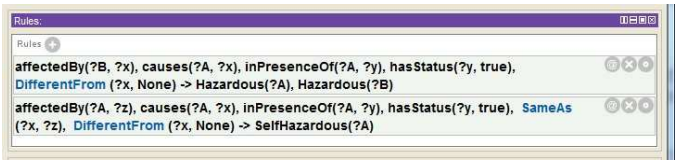

Figure A.7: SWRL Rules

We have used two SWRL rules to make selection of the proper objects and classify them as members of specific class. The first rule is used to classify all the item pairs that may have incompatibility considering if the the external conditions are favorable. In that case they asserted as members of the class $\mathrm{Hazardous}$. The first rule in fig A.7 performs this classification. The second rule verifies if an item is self hazardous with favorable external condition. If it's so, the item is asserted as member of class $\mathrm{SelfHazardous}$.

A.2 Applications

We have proposed the system using ontology as it's local knowledgebase to infer incompatibilities on the principle of InoT. We think that it can be used to infer incompatibilities among objects in various domains. "Bin That Thinks" is a project, that aims to propose an intelligent waste management solution based on item level identification. The goals are to improve recycling efficiency, reducing waste processing cost and avoiding hazardous situations [TA]. Though we have not assessed for the financial benefits figuratively for using our system, our approach hints at the benefits qualitatively. Sorting wastes at the earliest retains the purity of the recyclables. This reduces the cost of sorting at a later stage in processing plants by waste management companies like Veolia, which is usually passed on to the consumers as penalties on the cities.

We have developed an application for the domain of waste management using the system described above. It can be used to make inferences for incompatibilities and hazards among the waste items present collectively at a place. They may be situated inside a bin or a waste collecting vehicle or at the processing plant. For very complex domains like waste management, they are sometimes verified at every step in the processing chain. Alternatively, when the processing is performed at a single point, we consider the acceptance of error up to some limit. Fig A.8 shows the smart bin developed that can identify the RFID tagged wastes and

Figure A.8: Smart Bin

make inferences from its contents. In fig A.9 below shows a screenshot of our application. It shows the instance when an incompatibility is detected with two items present locally in the bin and the last item that was scanned. It also displays the inferred reasoning.

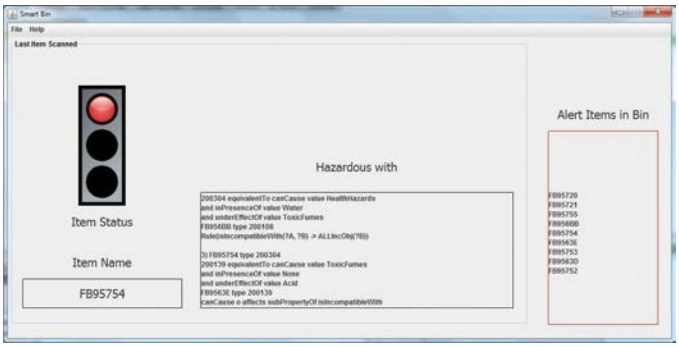

Figure A.9: Hazard Detection Application for Waste

Another domain of application for our model can be in the field of medicine. Storing medicines together can sometimes be potentially dangerous. It might also lead to confusion and take wrong medications. The elderly people and children are more vulnerable to such mistakes. Also some medicines might react with each other ('interact') if taken together and might cause serious problems.

A.3 Complexity

The OWL from the W3C has capabilities to describe concept from very simple to quite complex ones. It provides with a variety of features to express some domain of interest.

OWL ontology has has three different types of sub-languages namely OWL-Lite, OWL-DL and OWL-Full. These sub-languages differ in the amount of features incorporated in it and hence, have varying degrees of expressiveness. W3C provides a description of the features to be used for OWL-Lite compared to OWL-DL or OWL-Full [(W3a]. The profile for our model gets disqualified from being OWL-Lite as we have used OWL individuals to occur in descriptions or class axioms i.e. by using the value constraint owl:hasValue [Baa03, (W3a, (W3b]. The OWl-Lite has computational complexity of polynomial order whereas the rest grows exponentially [KR07]. So, it essentially means that the computational complexity of our model can be calculated to grow exponentially with increase in the ontology size. This is important in the context of pervasive computing as the setup would be functioning on embedded computers, which have limited memory and computational capabilities.

Appendix B

Publications

B.1 Conference Proceedings

- Arnab Sinha and Paul Couderc. Using owl ontologies for selective waste sorting and recycling. In 9th OWL: Experiences and Directions Workshop (OWLED 2012), Crete, Greece, 2012
- Arnab Sinha and Paul Couderc. Smart bin for incompatible waste items. In Proceedings of the 9th International Conference on Autonomic and Autonomous Systems (ICAS 2013), pages 40–45, Lisbon, Portugal, 2013
- Arnab Sinha and Paul Couderc. A framework for interacting smart objects. In ruSMART 2013 : The 6th conference on Internet of Things and Smart Spaces, Saint Petersburg, Russia, 2013. Springer
- Arnab Sinha, Yann Glouche, and Paul Couderc. Distributed tree structure for composite physical objects. In Proceedings of the 5th International Conference on Ambient Systems, Networks and Technologies (ANT 2014), Hasselt, Belgium, 2014. Elsivier

B.2 Journal

- Yann Glouche, Arnab Sinha, and Paul Couderc. A smart waste management with self-describing complex objects. 2014. (accepted, under revision by authors)

Appendix C

Résumé de la thèse en français

C.1 Introduction

Ces dernières années, l'attention s'est déplacée du modèle traditionnel d'interaction reposant sur la métaphore du bureau vers les tendances proposées par "l'informatique diffuse" et l'informatique "dans les nuages". Cette dernière a emergée comme un paradigme important et largement adopté par l'industrie. Les infrastructures supportant ce modèle comportent d'immenses puissances de calcul de capacités de stockage. L'extensibilité est également souvent "elastique", permettant d'adapter à la demande les ressources comme les capacités de traitement et le stockage. Ces infrastructures sont habituellement gérées et opérées par des tierses parties, et leurs services peuvent être utilisés très simplement (parfois même "inconsciemment") par de petites organisation ou des particuliers de n'importe où et à n'importe quel moment, avec des modèdes de revenus généralement associés à l'usage des ressources. De son coté, l'informatique diffuse cherche à intégrer les systèmes informatiques avec les entités du monde physique. Ces dernières sont "augmentées" avec des capacités numériques qui leur sont rattachées. Un des objectifs est d'assurer cette intégration de manière impercetible, de telle sorte que les traitement numériques puissent s'effectuer en accompagnant de manière naturelle (ou implicite) les activités réelles, sans intervention explicite des utilisateurs. Les objets physiques considérés peuvent être aussi triviaux que de simples déchets participant à une application d'informatique diffuse pour faciliter le traitement des ordures ménagères.

L'approche "nuage" présente de nombreux avantages en terme de flexibilité et d'accessibilité aux ressources. L'Internet des objets, tel qu'il est entendu classiquement, propose d'étendre le "nuage" aux appareils et objets usuels. Les équipements connectés offrent également aux utilisateurs la possibilité de profiter des services du nuage de manière ubiquitaire. Cependant, l'approche "nuage" comporte des aspects problématiques, qui peuvent constituer des défis difficiles pour certaines applications. La forte dépendance vis à vis du réseau peut constituer une faiblesse : en effet, la connectivité réseau est parfois instable et peut rendre le service inopérant. De plus, les applications impliquant de nombreuses entités et/ou évènements entrainent une charge réseau importante, qui doit être supporté par l'infrastructure, tant du point de vue du réseau que du traitement. Le coût financier et/ou énergétique de cette charge réseau peut également constituer une limitation. A coté des aspects techniques, on peut voir poindre

d'autres inquiétudes avec cette approche. En particulier, la centralisation massive de données aux mains de tierses parties est une menace pour la confidentialité et la vie privée. Enfin, l'ouverture d'accès à des données sensibles afin de les traiter dans le nuage peut être difficile ou interdite dans certains contextes.

Des approches alternatives existent en informatique diffuse, consistant à coupler étroitement les processus de traitement de l'information aux activités se déroulant dans le monde physique. Les objets (ou les entités physiques) sont "augmentés" avec des informations numériques, et lorsque des activités utilisent ces objets, les calculs ou les processus de traitement utilisent les données associées en parallèle. On notera la différence avec l'approche "nuage" précédante, où les traitements s'effectuent sur une représentation abstraite du monde réel (modèle) située dans le nuage. Nos travaux suivent l'approche diffuse où l'information et son traitement sont étroitement couplés aux objets physiques et leurs interactions réelles.

Dans cette thèse, nous avons étudié des approches pour stocker des données et structures de données directement sur des objets physiques, en s'appuyant sur la technologie RFID. La RFID est une technologie émergente qui est surtout utilisée pour lier un objet physique à un système d'information. Nous avons suivi une approche moins commune, qui consiste à utiliser le support RFID comme un espace mémoire pour nos structures de données. Cette approche permet en particulier de mettre en oeuvre la notion d'objet "auto-descriptif", où un objet (simple ou complexe) peut caractériser ses propriétés de manière autonome pour un système intelligent qui l'interroge. Les avantages par rapport à l'approche "nuage" sont de plusieurs ordre. Cette architecture reposant sur des interactions locales, il n'y a pas de dépendances au réseau ni à des infrastructures de traitement distante ; l'extensibilité est donc naturelle, et la robustesse plus simple à assurer. De plus, l'absence de centralisation de données associées aux entités physiques reduit considérablement les menaces que l'on peut craindre sur la confidentialité et la vie privée. De même, le déploiement de tels systèmes, par nature indépendants, est facilité du fait de l'absence de contrainte d'interfacage ou d'intéropérabilité avec des systèmes d'information existants.

Pour autant, cette approche comporte également ses défis propres. Ils sont principalement liés aux limitations technologiques de la RFID. Notamment, l'espace mémoire disponible sur les puces RFID est très réduit. Le codage des informations doit donc être aussi frugal que possible en consommation mémoire. Notre étude propose des solutions pour y répondre, pour des données sémantiques caractérisant des objets simples, et pour des structures de données caractérisant des objets complexes. Nous l'avons en particulier appliqué sur au domaine de la gestion des déchets, pour un système de conteneurs intelligents capables d'améliorer l'efficacité du tri amont des déchets. On réduit ainsi les produits indésirables au niveau des conteneurs et l'on améliore la qualité du tri. L'application que nous avons developpé permet également la detection immédiate d'incompatibilité entre un produit présenté et ceux déjà présents dans un conteneur. Cette application a été développée dans le cadre du projet BinThatThinks, une collaboration entre l'INRIA et deux partenaires industriel, Veolia et Etineo, financée par l'Agence Nationale pour la Recherche.

C.2 Etat de l'art

L'informatique diffuse, telle qu'imaginée par Mark Weiser, correspond à la tendance toujours plus forte d'intégration d'intelligence dans les objets usuels et l'environnement. L'idée fondatrice reste identique, même si des variantes terminologiques sont apparues comme intelligence ambiante ou intelligence ubiquitaire [Gre10], pour refleter certaines différences légères. D'autres termes sont également utilisés comme informatique physique, the Internet des objets [1], technologie haptique [Sad07], and objets intelligents [2], mettant l'accent sur les objets physiques impliqués [HPT97].

Une approche particulière en informatique diffuse repose sur l'idée d'un couplage étroit entre le monde réel et les systèmes d'information. Les objets et entités du monde physiques constituent directement l'interface d'usage du système numérique, comme c'est de rigueur selon le principe de l'informatique ubiquitaire. Mais de plus, les processus de traitement de l'information, ou de calcul, sont eux aussi directement supportés par les activités physiques. Considérons l'exemple d'une personne effectuant des achats dans un supermarché. Les produits constituent directement des données (les prix des produits), et le coût total du caddy peut être calculé directement par ce dernier à mesure que des produits sont ajoutés ou retirés.

La figure C.1 montre les trois éléments principaux d'un système diffus : d'abord, l'environnement physique, constitué des entités réelles (objets, personnes, lieux etc). Ensuit, l'environnement numérique, ou virtuel, constitué par des systèmes d'information comme l'Internet. Le dernier groupe est un ensemble d'interfaces permettant de coupler les deux environnements précédant. Idéalement, les interfaces homme-machine explicite, comme les interfaces graphiques, devraient disparaitre car dans la vision ubiquitaires les activités réelles constituent elles mêmes l'interface de contrôle de l'environnement numérique.

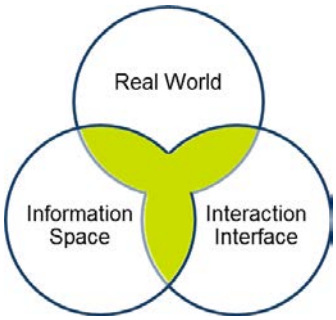

FIGURE C.1 – Elements d'un système d'informatique diffuse

L'informatique diffuse est mise en oeuvre grâce à des systèmes embarqués et des capteurs disposés dans l'environnement. Ceux-ci collectent des données permettant de caractériser la situation physique. Ces données sont remontées vers des systèmes de traitement, afin de les analyser pour en extraire une représentation de haut niveau (contexte), exploitable par les applications. Une infrastructure de communication est nécessaire pour permettre cette remontée

1. http://www.rfidjournal.com/articles/view?4986
2. http://ttt.media.mit.edu/vision/vision.html

d'information.

Une qualité essentielle de l'informatique étant sa discretion et son caractère "implicite", elle fait naturellement appel à des systèmes de communication sans fil, et plus précisement à deux types d'infrastructures. D'une part, les réseaux à couverture globable, comme les réseaux cellulaires, sont utilisés pour accéder à des ressources numériques distantes. D'autre part, l'informatique diffuse exploite de manière particulière les communications de proximité (ou communication ad hoc) pour supporter des interactions liées au contexte physique. Les deux types d'infrastructures peuvent êtres utilisées seules, ou en combinaison, en fonction des contraintes applicatives. Le symbole étoile dans la table 1.9 montre comment la transition des usages s'effectue dans le contexte des applications d'informatique diffuse. Les communications globables sont principalement utilisée pour les echanges de données machines à machines, avec des débits importants et indépendamment de la localisation. A mesure qu'on descend vers les communications plus localisées, le rôle de transmission de donnée s'amoindrit au profit du lien avec le contexte physique. Sur cet aspect, l'infrarouge et la RFID constituent deux exemples particulièrement appropriés.

La figure C.2 représente de manière visuelle la transition d'usage entre les deux types de communication.

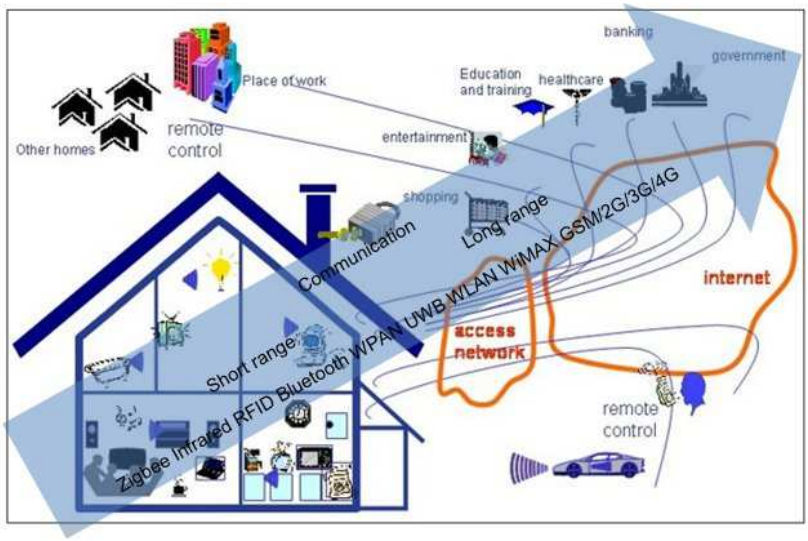

FIGURE C.2 – Communications globales et communications à courte portée

Les applications de l'informatique diffuse s'appuyant largement sur les communications de proximité pour établir le lien avec le contexte physique, il est important d'examiner leurs propriétés. La table C.1 en dresse un résumé. On notera que la nature de la propagation peut être utilisée pour concevoir précisement la couverture spatiale nécessaire pour certaines interactions.

On peut citer différents travaux illustrant ces aspects : [May, KhPV+13, BdS08, NSS13].

	WPAN	Bluetooth	Zigbee	Infrared	UWB	RFID
Identification	X	X	X	X	X	X
Addressing mode	Multiple	Multiple	Multiple	Individual	Both	Both
Propagation	Non-line-of-sight (NLOS)	NLOS	NLOS	Line-of-sight	NLOS	NLOS
Range	1m-100m	upto 10m	10m-100m	upto 10m	upto 30m	upto 10m

TABLE C.1 – Propriétés : communications courte portée

La localisation par RFID [WHFG92]. The "Tile"™ [3] est un produit commercial qui exploite ce concept.

La RFID est une technologie particulière parmi les systèmes de communciation à courte portée, en quelque sorte hybride entre technologie de communication et technologie de perception. Elle est principalement utilisée pour lié des informations numérique à des objets physique à travers un identifiant, mais plus généralement, elle permet de stocker une petite quantité d'information arbitraire sur un objet physique. On peut donc la considérer comme un petit espace mémoire associé à un objet, et adressable à proximité par un dispositif approprié.

Il existe très peu de travaux dans la literature qui exploite cette dernière caractéristique, autorisant pourtant des traitements autonomes ou décentralisés, sur des objets individuels ou sur des groupes d'objets. Considérons l'exemple de produits périssables étiquettés, traversant une chaine du froid où sont journalisés les paramètres environnementaux importants (temperature notamment) à chaque étape. Un objets affecté à une étape par des conditions anormales peut ainsi être "marqué" et vérifié à n'importe quand, et de manière autonome. Cette application est démontrée dans [YL09]. La vérification d'intégrité d'objets ou de groupes d'objets est une autre application où l'usage d'une architecture RFID décentralisée a été démontrée [CBA11]. L'espace mémoire des objets étiquettés est utilisé pour stocker des données d'intégrité sur le groupe, et ainsi supporter le contrôle d'intégrité. Ce concept a également été démontré dans l'application Ubi-Post, pour sécuriser le transport et la livraison d'objets de valeurs [VBC12].

Les deux formes d'approches, centralisée et distribuée, sont comparées par Diekmann [DMS07]. Elles sont appelées respectivement données sur la puce (data-on-tag) et données dans le réseau (data-on-network). Au delà des avantages et inconvénients de ces deux approches, ce travail suggère une approche hybride permettant d'améliorer la disponibilité des données pour les traitements. La figure C.3 présente les différences. Les fleches pleines représentent les requêtes d'accès aux données indispensables et les fleches pointillées les requêtes à des données optionnelles. En C.3(a), on montre les étiquettes qui ont une dépendance systématique aux bases de données distantes. Les cas C.3(b) et C.3(c) representent deux variantes d'un modèle autonome, où l'espace mémoire des étiquettes stocke des données avec un accès optionnel à la base de donnée distante. Le cas (b) represente des étiquettes stockant des des données individuelle, tandis que sur le cas (c) les données représentent des données liées entre elles.

3. Tile, the world's largest lost and found. `http://www.thetileapp.com/`

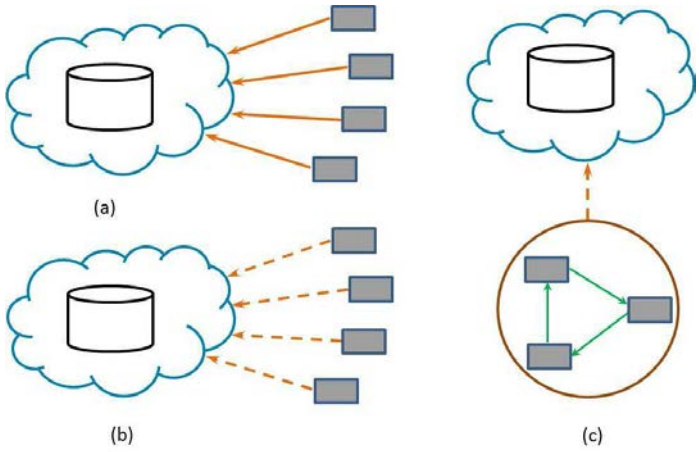

FIGURE C.3 – Architectures de traitement de données

C.3 Objets intelligents avec des données tangibles

Ce travail de thèse s'intègre dans le cadre du projet BinThatThinks, qui étudiait des solutions innovantes pour de futures chaines intelligente de traitement de déchets. Nos contributions proposent des solutions pour exploiter l'espace mémoire des étiquettes RFID en tant que support pour des données et structures de données distribuées associées aux objets, autorisant des traitements autonomes, c'est-à-dire sans dépendance vis-à-vis d'infrastructures externes. L'un des principaux défis est l'espace mémoire très restreint disponible dans les étiquettes RFID. Le reste de cette section explique plus en détails nos deux contributions principale, en commencant par la notion d'objet auto-descriptif, puis celle de structure de données tangibles.

Nous proposons le principe d'objets physiques auto descriptifs dans l'objectif d'améliorer l'autonomie d'application diffuse qui les utilisent. Traditionnellement, les informations associées à un objets, qu'elles soient destinées à l'usage du fabriquant, des distributeurs ou de l'utilisateur final, sont situées dans des systèmes d'information externes. Le lien peut être assuré avec l'objet physique avec une référence, electronique ou non. En stockant certaines informations sémantiques, comme les propriétés de l'objet, directement sur l'objet lui même, sous avons montré qu'il est possible de construire des applications sensibles au contexte capables de prendre des décisions de manière autonomes. Notons que cette approche n'est pas incompatibles avec les usages traditionnels de la RFID, c'est-à-dire de stocker une référence vers un système d'information externe. Notre approche offre les avantages suivants :

- Disponibilité en l'absence d'infrastructure réseau (ou en cas de défaillance) et passage à l'echelle item Facilité d'intégration dans des systèmes existants et d'interopérabilité
- Meilleures garanties en termes de respect de la vie privée, et de securité

En effet, dans les architectures plus classiques, on a généralement recourt à une infrastructure de capteurs collectant des mesures de bas niveau, qui doivent être remontées puis interpréter par analyse pour en dériver la représentation de haut niveau (contexte). En évitant ces étapes du fait de la disponibilité directe des éléments caractérisant le contexte sur les objets, on

simplifie grandement l'architecture et l'on supprime les difficultés liées à la génération de messages vers l'infrastructure et à la concentration de grandes masses de données, potentiellement sensibles, sur des serveurs contrôlés par des tiers.

Plus spécifiquement, le principe d'objets physiques auto descriptifs repose sur la notion de "données tangibles", qui constituent une caractérisation sémantique de l'objet ou de ses propriétés vis-à-vis d'un contexte de traitement. La représentation de ces informations sur le support très contraint des RFID nécessite de trouver le bon compromis entre l'expressivité nécessaire pour l'application, et la consommation mémoire que cela entraine. Nous avons proposé certaines solutions pour s'accomoder à ces contraintes.

L'approche a été validée dans le cadre d'une application mise en oeuvre sur le protoype BinThatThinks, pour la gestion de déchets. L'application facilite le tri précoce des déchets au moment de leur dépose, pour faciliter le recyclage. Chaque produit dispose d'une auto-description, caractérisant sa nature, les proportions de ses consistuants. De l'autre coté, des conteneurs intelligents sont capables d'identifier les produits qui leurs sont présentés, et de les accepter ou de les refuser en fonction du type de déchets collectés, mais aussi de potentielles interactions avec le contenu déjà présent. L'objectif est de maximiser les possibilités de recyclage, en améliorant la qualité des produits retournés, en évitant la présence d'indésirables ou de contaminants. En effet, l'ensemble accumulé dans un conteneur peut être rendu impropre au recyclage à cause d'un élément contaminant. C'est par exemple le cas en introduisant une bouteille en verre dans un conteneur destiné à collecter des bouteilles en plastique [onl]. Il peut également se produire des situations potentiellement dangereuses avec certains éléments, dans certaines conditions. Les bouteilles d'aérosols, ou de produits inflammables peuvent provoquer dans des combinaisons défavorables des départ de feu ou même des explositions. Notre application exploite également l'auto-description des objets pour prévenir ces situations.

Bien que l'on puisse espérer un accroissement de la capacité de stockage des étiquettes RFID avec le progrès de la technologie, la nature passive des étiquettes (donc alimentation en énergie épémère, par le lecteur) risque de rendre pendant longtemps cette capacité limitée. Cependant, on peut constater que souvent, que plus la description d'un objet est complexe, plus l'objet physique concerné est lui même "complexe", c'est-à-dire composé de plusieurs éléments ou groupes d'éléments. Il est possible de profiter de cette constatation, en répartissant la description de l'objet complexe sur des différents éléments. Nous introduisons la notion de structure de donnée tangibles, précisement destinée à ces situations. Dans la suite, nous expliquons cette approche et la démontrons sur diverses applications.

Il est très commun de rencontrer des groupes d'objets apparentés, ou un objet complexe constitué de sous-éléments. Si chaque objet ou élément possède une étiquette, on peut allouer tout ou partie de l'espace disponible dans chaque élément pour y stocker des informations pertinentes sur le groupe, ou sur les relations des éléments entre eux. Il est ainsi possbile de construire des structures de données pour supporter notamment le tri ou l'arrangement des objets, la vérification de l'intégrité d'un groupe, l'assemblage ou le desassemblage etc. Typiquement, la structure de donnée refletera les propriétés physiques selon lesquelles l'objet complexe est construit ou composé. C'est ce qui motive l'appelation de structure de données tangibles. Considérons l'exemple d'une bibiothèque. Si les données tangibles de chaque livre renseignent les livres adjacents, la séquence physique est aussi représentée par la structure de donnée tangible, comme le montre la figure C.4 où les liens vers les livres adjacents sont représentés par

les fleches.

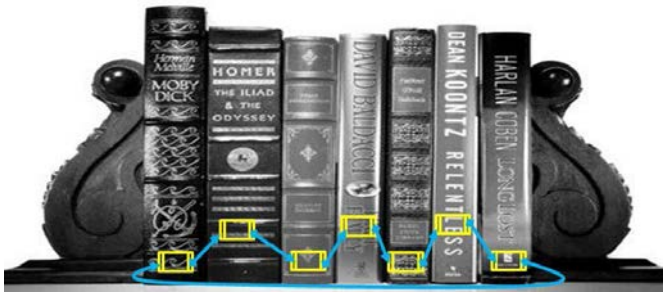

FIGURE C.4 – Bibliothèque d'objets reliés (livres)

Un téléphone mobile peut être considéré comme un objet complexe, comme montré sur la figure figure C.5. Imaginons que ses composants importants (et coûteux) soient étiquettés.

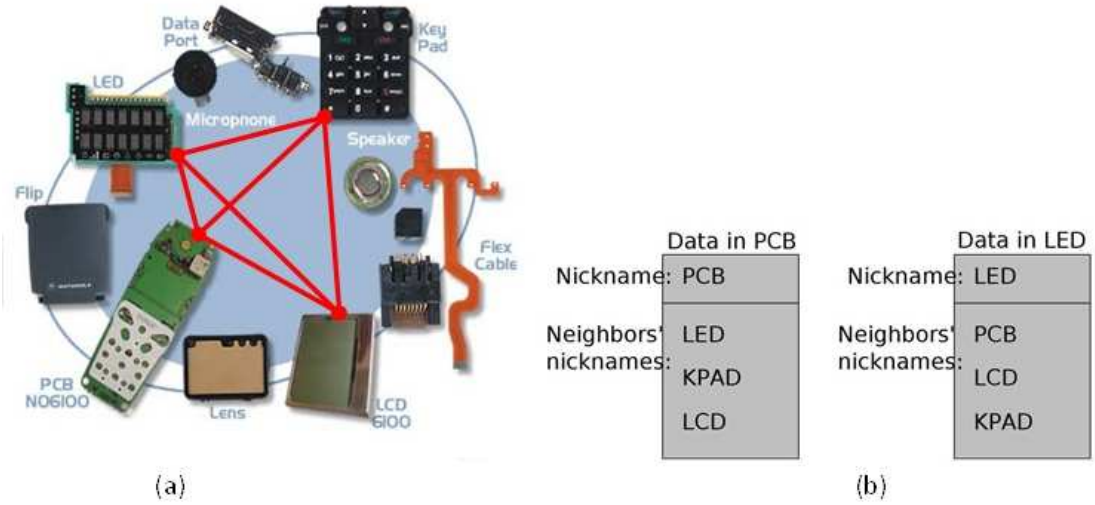

FIGURE C.5 – Objets liés exploitant un graphe complet

C.4 Conclusion et Perspectives

L'informatique diffuse est une tendance qui succède à l'ère du "poste de travail" qui prévalait précédemment, où les ressources manipulées étaient représentés selon une métaphore graphique par des symboles du monde réel. On pouvait observer un séparation claire entre les environnements réels et numériques, et l'interface graphique établissait le lien. En informatique diffuse ce lien tend à disparaitre, grâce à une intégration massive et discrete de l'intelligence

FIGURE C.6 – Représentation composite d'une chaise sous la forme d'une hiérarchie

directement dans l'environnement réel. Cependant, l'emergence de ce nouveau paradigm s'accompagne de défis importants, en particulier du fait de la dépendance aux infrastructures de communications globales et de traitement de l'information centralisé.

Dans cette thèse, nous avons proposé une approche différente en ce qui concerne la mise en oeuvre de l'informatique diffuse. Les données de haut niveau pour décrire le contexte des applications sont directement stockés sur les objets, grâce à des étiquettes RFID, permettant un traitement local et autonome de l'information. On évite ainsi plusieurs problèmes importants associés aux infrastructures globales et à l'approche "nuage", comme les menaces pour la vie privée.

En première approche, nous avons proposé une "auto-description" d'objets simples, indépendemment les uns des autres, avec des données dites "tangibles", stockées dans des étiquettes RFID. Cela permet d'intégrer, en tant qu'entités numériques, la plupart des objets courants dans des applications ou services d'informatique diffuse et de les faire participer aux interactions des environnements intelligents. Les "données tangibles" décrivant ces objets peuvent être des status d'état, des propriétés qualitative ou quantitative de l'objet et plus généralement toute information utile pour permettre à l'objet de participer à des interactions avec l'environnement. Un exemple de donnée tangible peut être une information concernant le recyclage d'un produit. Un défi important pour la mise en oeuvre de cette approche est la quantité de mémoire très limitée offerte par les étiquettes RFID. Aussi, il est essentiel de bien selectionner les données pertinentes à placer à ce niveau, et de les encoder de manière efficace. Nous avons démontré les avantages de plusieurs schémas de codage exploitant des caractères décimaux ou hexadécimaux dans le cas d'une application de tri selectif pour la gestion de déchets. Ils sont adaptés en fonction des informations à décrire.

Nous avons également montré comment l'usage d'un standard existant (dans le domaine des déchets) et la ré-utilisation des mêmes données à différents stades du cycle de vie d'un

produit permet de réduire les informations redondantes, et donc la quantité de mémoire nécessaires aux objets auto-descriptifs dans différents services. Ainsi, on réduit à la fois l'impact de la mémoire réduite des étiquettes RFID, tout en favorisant l'interopérabilité grâce aux données partagées par les différents services.

La nature autonome de cette approche réduit la dépendance vis-à-vis d'infrastructures externes, ce qui peut être un avantage en terme de disponibilité et d'extensibilité : en effet, traditionnellement, la disponibilité et l'extensibilité doivent être envisagés et dimensionnés selon le nombre totale d'entités participant au système, alors que dans notre approche les traitements étant effectués localement, la charge de traitement ou de trafic réseau n'augmente pas le nombre d'entités.

De plus, on évite également la concentration de données, potentiellement sensibles d'un point de vue sécurité ou vie privée, sous le contrôle central de tierses parties, ce qui est facilement le cas lorsque les étiquettes RFID se limitent à des identifiants ou des pointeurs vers des données externes ou des services "nuages". En particulier, il serait aisé pour ceux disposant de ces données externes d'établir des profiles comportementaux précis grâce à l'analyse des accès à ces données.

Enfin, le coût réduit constitue également un avantage de cette approche. Les étiquettes en elles-même permettent d'étendre les objets courant pour quelques centimes (à quelques dizaines de centimes) d'euros, et surtout en évitant la dépendance à des infrastructures externe, on n'a pas à supporter de coût d'accès ou d'usage à ces infrastructures (réseau, ou de traitement).

En seconde approche, nous avons appliqué le principe des données tangibles à des objets multiples (ou groupes d'objets). En effet, dans de nombreuses situations des objets physiques sont des entités complexes, composées de multiples éléments ou tout du moins en relation avec d'autres éléments ou objets physiques. En s'appuyant sur des étiquettes réparties sur les différents éléments constituant un groupe d'objets ou un objet complexe, il est possible de construire un système d'information autonome pour décrire les propriétés de l'objet composite résultant. De plus, dans cette approche la mémoire utilisée pour le stockage est une répartie sur de multiples étiquettes (attachées aux éléments), réduisant la contrainte de la mémoire très limitée lorsqu'on ne dispose que d'une étiquette unique. Nous avons montré comment construire des structures de données distribuées physiquement selon cette architecture, permettant de refleter les relations physiques entre les objets. De telles structures, que nous appelons structures de données tangibles, supportent la mise en oeuvre d'applications ou services pervasif comme la vérification de l'intégrité d'un système (et l'identification d'éléments manquants), l'assistance à l'assemblage ou à la maintenance. Ici encore, l'approche reduit les menaces qu'on peut craindre pour la vie privée lorsque des information liées sont fortement concentrées. Nous avons démontré cette seconde approche dans une application d'assistance à l'assemblage d'objets physiques complexes.

Dans cette thèse, nous avons défendu une approche consistant à attacher des "données tangibles" sur les objets physiques eux-mêmes pour les rendre "intelligents", grâce à des étiquettes RFID. Cette technologie, bien que prometteuse, comporte de nombreuses limitations, en particulier au niveau de la capacité mémoire. Notre approche ouvre une nouvelle direction en rupture avec les architectures habituellement utilisées, qui consistent à s'appuyer sur une infrastructure externe. Notre approche offre en l'avantage notable de réduire les risques asso-

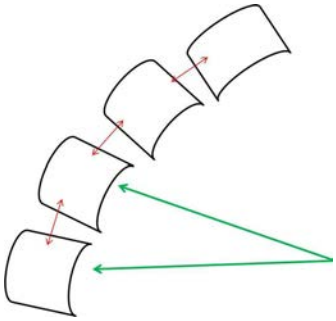

FIGURE C.7 – Inventaire sectoriel d'une bibliothèque avec un lecteur RFID à antenne intelligente

ciés à la sécurité et au respect de la vie privée. Les données sémantiques restent "diffuses" puisqu'elles sont physiquement associées aux objets, et toujours directement accessibles pour des traitements dans l'environnement de l'objet. Les attributs sémantiques du contexte peuvent directement être exprimés par les objets, sans infrastructure de perception ni traitement d'interprétation ; en quelque sorte, on peut dire que l'approche offre une interface sémantique "native" aux objets physique, sans couche intermédiaire. Des opérations analogues aux traitements numériques comme les recherches, les tris etc. peuvent s'effectuer directement dans l'environnement. Thingful est un service lancé récemment par une start-up qui espère opérer le même type de service que Google ® mais sur l'Internet des objets [Thi].

En dehors des restrictions mémoire, la partie transmission de la RFID doit également progresser pour être capable d'adresser de manière fiable l'ensemble des objets situés à proximité des antennes de lecture. Pour le moment, les caractéristiques du champs (puissance, direction, polarité) peuvent être ajustées, et des contrôleurs intelligents sont développés pour offrir des infrastructures plus fiables.

Une perspective serait d'exploitant de nouveaux lecteurs à champs sectorisés [RFI], permettant des lectures selectives de zone de l'espace. En s'appuyant sur cette technologie, des structures de données tangibles s'étendant sur des volumes ou des zones importants pourraient être mis en oeuvre, et ainsi permettre par exemple l'auto-description d'un environnement (pièce, bâtiment...), ou encore de collection d'objets répartis dans un espace important. La figure C.7 montre comment les livres d'une bibliothèque reliés par une liste pourraient être lus avec ce type de lecteurs.

Au delà des progrès technologiques dont l'approche que nous avons défendu pourrait bénéficier, une autre perspective concerne la conception des structures de données tangibles : actuellement, les structures de données tangibles que nous avons proposé doivent être elaborées pour chaque cas, en fonction des contraintes de l'application, par le concepteur. Un outil qui permettrait d'assister la conception de telles structures serait une aide importante, notamment pour faciliter le compromis à établir entre résilience et consommation mémoire.

Bibliography

[AMS02] Stavros Antifakos, Florian Michahelles, and Bernt Schiele. Proactive instructions for furniture assembly. In UbiComp 2002: Ubiquitous Computing, pages 351–360. Springer, 2002.

[AOH07] Mohammadreza Asghari Oskoei and Huosheng Hu. Myoelectric control systemsŮa survey. Biomedical Signal Processing and Control, 2(4):275–294, 2007.

[APN+09] E. Abad, F. Palacio, M. Nuin, A. González de Zárate, A. Juarros, J.M. Gómez, and S. Marco. {RFID} smart tag for traceability and cold chain monitoring of foods: Demonstration in an intercontinental fresh fish logistic chain. Journal of Food Engineering, 93(4):394 – 399, 2009.

[Ass] IEEE Standards Association. IEEE 802.11™: Wireless LANs @ONLINE. Available at `http://standards.ieee.org/about/get/802/802.11.html` [accessed 14 September 2013,16h15].

[AWL+04] Urs Anliker, Jamie A Ward, Paul Lukowicz, Gerhard Troster, Francois Dolveck, Michel Baer, Fatou Keita, Eran B Schenker, Fabrizio Catarsi, Luca Coluccini, et al. Amon: a wearable multiparameter medical monitoring and alert system. Information Technology in Biomedicine, IEEE Transactions on, 8(4):415–427, 2004.

[Baa03] Franz Baader. The description logic handbook: theory, implementation, and applications. Cambridge university press, 2003.

[BAC09] Michel Banâtre, Fabien Allard, and Paul Couderc. Ubi-check: A pervasive integrity checking system. In Smart Spaces and Next Generation Wired/Wireless Networking, pages 89–96. Springer, 2009.

[BBS+08] Stacy Bamberg, Ari Y Benbasat, Donna M Scarborough, David E Krebs, and Joseph A Paradiso. Gait analysis using a shoe-integrated wireless sensor system. Information Technology in Biomedicine, IEEE Transactions on, 12(4):413–423, 2008.

[BdJHS12] Ralph Barthel, Martin de Jode, and Andrew Hudson-Smith. Approaches to interacting with digital object memories in the real world. In Proceedings of the 2012 ACM Conference on Ubiquitous Computing, UbiComp '12, pages 1179–1182, New York, NY, USA, 2012. ACM.

[BdS08] M. Bouet and A.L. dos Santos. Rfid tags: Positioning principles and localization techniques. In Wireless Days, 2008. WD '08. 1st IFIP, pages 1–5, 2008.

[BE07] Leah Buechley and Mike Eisenberg. Boda blocks: a collaborative tool for exploring tangible three-dimensional cellular automata. In Proceedings of the 8th iternational conference on Computer supported collaborative learning, CSCL'07, pages 102–104. International Society of the Learning Sciences, 2007.

[BFO92] Michael Bajura, Henry Fuchs, and Ryutarou Ohbuchi. Merging virtual objects with the real world: seeing ultrasound imagery within the patient. In Proceedings of the 19th annual conference on Computer graphics and interactive techniques, SIGGRAPH '92, pages 203–210, New York, NY, USA, 1992. ACM.

[bin] Bin That Thinks prototype demonstration video. Available at `http://www.youtube.com/watch?v=daac0tFig34` [accessed 26 July 2014,17h30].

[BKH13] Ralph Barthel, Alexander Kröner, and Jens Haupert. Mobile interactions with digital object memories. Pervasive and Mobile Computing, 9(2):281 – 294, 2013. Special Section: Mobile Interactions with the Real World.

[BM01] Richard D Beach and Francis Moussy. Totally implantable real-time in vivo video telemetry monitoring system for implant biocompatibility studies. Instrumentation and Measurement, IEEE Transactions on, 50(3):716–723, 2001.

[CAJ09] Diane J. Cook, Juan C. Augusto, and Vikramaditya R. Jakkula. Ambient intelligence: Technologies, applications, and opportunities. Pervasive and Mobile Computing, 5(4):277 – 298, 2009.

[CBA] Rohit Chaudhri, Gaetano Borriello, and Richard Anderson. Pervasive computing technologies to monitor vaccine cold chains in developing countries.

[CBA11] Paul Couderc, Michel Banâtre, and Fabien Allard. Pervasive integrity checking with coupled objects. Procedia Computer Science, 5(0):320 – 327, 2011. The 2nd International Conference on Ambient Systems, Networks and Technologies (ANT-2011) / The 8th International Conference on Mobile Web Information Systems (MobiWIS 2011).

[CC07] B. Chowdhury and M.U. Chowdhury. Rfid-based real-time smart waste management system. In Telecommunication Networks and Applications Conference, 2007. ATNAC 2007. Australasian, pages 175–180, 2007.

[CDBN09] Andrea Caragliu, Chiara Del Bo, and Peter Nijkamp. Smart cities in europe. 2009.

[CDK01] George Coulouris, Jean Dollimore, and Tim Kindberg. Distributed systems (3rd ed.): concepts and design. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2001.

[CEF+12] Marie Chan, Daniel Estève, Jean-Yves Fourniols, Christophe Escriba, and Eric Campo. Smart wearable systems: Current status and future challenges. Artificial Intelligence in Medicine, 56(3):137 – 156, 2012.

[CMD11] Gastone Ciuti, Arianna Menciassi, and Paolo Dario. Capsule endoscopy: from current achievements to open challenges. Biomedical Engineering, IEEE Reviews in, 4:59–72, 2011.

[CMS10] Ayesha Chaudhary, Michael J McShane, and Rohit Srivastava. Glucose response of dissolved-core alginate microspheres: towards a continuous glucose biosensor. Analyst, 135(10):2620–2628, 2010.

[CNR] Features of rfid tags. Available at `http://www.centrenational-rfid.com/features-of-rfid-tags-article-19-gb-ruid-202.html` [accessed 08 October 2013,19h41].

[Com] European Commission. The EU's approach to waste management. Available at `http://ec.europa.eu/environment/waste/index.htm` [accessed 25 November 2013,10h58].

[Cou07] Joseph F Coughlin. Disruptive demographics, design, and the future of everyday environments. Design Management Review, 18(2):53–59, 2007.

[CP08] Joseph F Coughlin and James Pope. Innovations in health, wellness, and aging-in-place. Engineering in Medicine and Biology Magazine, IEEE, 27(4):47–52, 2008.

[CTCC03] Sean Clark, Ken Traub, Dipan Anarkat Uniform Code Council, and Ted Osinski Uniform Code Council. Auto-id savant specification 1.0 2. 2003.

[DBS+01] Ken Ducatel, Marc Bogdanowicz, Fabiana Scapolo, Jos Leijten, and Jean-Claude Burgelman. Scenarios for ambient intelligence in 2010. Office for official publications of the European Communities, 2001.

[DMS07] Thomas Diekmann, Adam Melski, and Matthias Schumann. Data-on-network vs. data-on-tag: managing data in complex rfid environments. In System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on, pages 224a–224a. IEEE, 2007.

[DTB+10] Jiangpeng Dai, Jin Teng, Xiaole Bai, Zhaohui Shen, and Dong Xuan. Mobile phone based drunk driving detection. In Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2010 4th International Conference on-NO PERMISSIONS, pages 1–8. IEEE, 2010.

[EB00] B.G. Evans and K. Baughan. Visions of 4G. Electronics Communication Engineering Journal, 12(6):293–303, 2000.

[ECPS02] Deborah Estrin, David Culler, Kris Pister, and Gaurav Sukhatme. Connecting the physical world with pervasive networks. Pervasive Computing, IEEE, 1(1):59–69, 2002.

[EU:] European waste catalogue and hazardous waste list. Available at `www.environ.ie/en/Publications/Environment/Waste/WEEE/` [accessed 08 December 2013,10h45].

[FFHR07] William H Fissell, Aaron J Fleischman, H David Humes, and Shuvo Roy. Development of continuous implantable renal replacement: past and future. Translational Research, 150(6):327–336, 2007.

[FIB95] George W. Fitzmaurice, Hiroshi Ishii, and William A. S. Buxton. Bricks: laying the foundations for graspable user interfaces. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '95, pages 442–449, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.

[Fit93] George W Fitzmaurice. Situated information spaces and spatially aware palmtop computers. Communications of the ACM, 36(7):39–49, 1993.

[FKK09] Daniel Fitton, Fahim Kawsar, and Gerd Kortuem. Exploring the design of a memory model for smart objects. In Intelligent Environments (Workshops), pages 33–38, 2009.

[Fle02] Richard Ribon Fletcher. Low-Cost Electromagnetic Tagging: Design and Implementation. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, July 2002.

[FLS03] Christian Floerkemeier, Matthias Lampe, and Thomas Schoch. The smart box concept for ubiquitous computing environments. In Smart Objects Conference, Grenoble, 2003.

[FMHS93] Steven Feiner, Blair MacIntyre, Marcus Haupt, and Eliot Solomon. Windows on the world: 2d windows for 3d augmented reality. In Proceedings of the 6th annual ACM symposium on User interface software and technology, UIST '93, pages 145–155, New York, NY, USA, 1993. ACM.

[FMHW97] Steven Feiner, Blair MacIntyre, Tobias Höllerer, and Anthony Webster. A touring machine: Prototyping 3d mobile augmented reality systems for exploring the urban environment. Personal Technologies, 1(4):208–217, 1997.

[FMS92] Steven Feiner, Blair MacIntyre, and Dorée Seligmann. Annotating the real world with knowledge-based graphics on a see-through head-mounted display. In Proceedings of the conference on Graphics interface '92, pages 78–85, San Francisco, CA, USA, 1992. Morgan Kaufmann Publishers Inc.

[FMS93a] Steven Feiner, Blair Macintyre, and Dorée Seligmann. Knowledge-based augmented reality. Communications of the ACM, 36(7):53–62, 1993.

[FMS93b] Steven Feiner, Blair Macintyre, and Dorée Seligmann. Knowledge-based augmented reality. Commun. ACM, 36(7):53–62, July 1993.

[FS91] Steven Feiner and Ari Shamash. Hybrid user interfaces: breeding virtually bigger interfaces for physically smaller computers. In Proceedings of the 4th annual ACM symposium on User interface software and technology, UIST '91, pages 9–17, New York, NY, USA, 1991. ACM.

[G+93] Thomas R Gruber et al. A translation approach to portable ontology specifications. Knowledge acquisition, 5(2):199–220, 1993.

[GC12] Yann Glouche and Paul Couderc. An autonomous traceability mechanism for a group of rfid tags. In UBICOMM 2012, The Sixth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, pages 161–168, 2012.

[GC13] Yann Glouche and Paul Couderc. A smart waste management with self-describing objects. In SMART 2013, The Second International Conference on Smart Systems, Devices and Technologies, 2013.

[GKdPC10] Sébastien Gambs, Marc-Olivier Killijian, and Miguel Núñez del Prado Cortez. Show me how you move and i will tell you who you are. In Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Security and Privacy in GIS and LBS, SPRINGL '10, pages 34–41, New York, NY, USA, 2010. ACM.

[Gre10] Adam Greenfield. Everyware: The dawning age of ubiquitous computing. New Riders, 2010.

[Groa] Bluetooth Special Interest Group. Bluetooth @ONLINE. Available at `http://www.bluetooth.org` [accessed 16 September 2013,11h49].

[Grob] IEEE Task Group. IEEE 802.15 WPAN Task Group 1 (TG1) @ONLINE. Available at `http://www.ieee802.org/15/pub/TG1.html` [accessed 16 September 2013,11h47].

[Groc] IEEE Task Group. IEEE 802.15 WPAN Task Group 4 (TG4) @ONLINE. Available at `http://www.ieee802.org/15/pub/TG4.html` [accessed 16 September 2013,17h36].

[GSC14] Yann Glouche, Arnab Sinha, and Paul Couderc. A smart waste management with self-describing complex objects. 2014. (accepted, under revision by authors).

[GTL+09] Toni Giorgino, Paolo Tormene, Federico Lorussi, Danilo De Rossi, and Silvana Quaglini. Sensor evaluation for wearable strain gauges in neurological rehabilitation. Neural Systems and Rehabilitation Engineering, IEEE Transactions on, 17(4):409–415, 2009.

[HB06] Eva Hornecker and Jacob Buur. Getting a grip on tangible interaction: a framework on physical space and social interaction. In Proceedings of the SIGCHI conference on Human Factors in computing systems, pages 437–446. ACM, 2006.

[HDT+08] Rasmus G Haahr, Sune Duun, Erik V Thomsen, Karsten Hoppe, and Jens Branebjerg. A wearable Şelectronic patchŤ for wireless continuous monitoring of chronically diseased patients. In Medical Devices and Biosensors, 2008. ISSS-MDBS 2008. 5th International Summer School and Symposium on, pages 66–70. IEEE, 2008.

[Hec10] Dominik Heckmann. User models meet digital object memories in the internet of things. In 18th Intl. Workshop on Personalization and Recommendation on the Web and Beyond, 2010.

[HMWC12] Amy J Halliday, Simon E Moulton, Gordon G Wallace, and Mark J Cook. Novel methods of antiepileptic drug deliveryŮpolymer-based implants. Advanced drug delivery reviews, 64(10):953–964, 2012.

[Hor08] Michael S. Horn. Tangible computer programming for informal science learning. In Proceedings of the 7th international conference on Interaction design and children, IDC '08, pages 21–24, New York, NY, USA, 2008. ACM.

[HPT97] Michael Hawley, R.Dunbar Poor, and Manish Tuteja. Things that think. Personal Technologies, 1(1):13–20, 1997.

[Huo99] H. Huomo. Fourth-generation mobile. ACTS Mobile Summit99, Sorrento, Italy, 1999.

[IEI] Rfid technology: Introduction and application. Available at `http://www.ieimobile.com/index.php?option=com_phocadownload&view=category&id=55&Itemid=101` [accessed 08 October 2013,19h36].

[IrD] IrDA. Infrared Data Association @ONLINE. Available at `http://www.irda.org/` [accessed 17 September 2013,11h40].

[Ish08] Hiroshi Ishii. Tangible bits: beyond pixels. In Proceedings of the 2nd international conference on Tangible and embedded interaction, TEI '08, pages xv–xxv, New York, NY, USA, 2008. ACM.

[IU97] Hiroshi Ishii and Brygg Ullmer. Tangible bits: towards seamless interfaces between people, bits and atoms. In Proceedings of the ACM SIGCHI Conference on Human factors in computing systems, CHI '97, pages 234–241, New York, NY, USA, 1997. ACM.

[JDCP10] Philippe Jourand, Hans De Clercq, and Robert Puers. Robust monitoring of vital signs integrated in textile. Sensors and Actuators A: Physical, 161(1):288–296, 2010.

[Kal07] A.T. Kalghatgi. Challenges in the design of an impulse radio based ultra wide band transceiver. In Signal Processing, Communications and Networking, 2007. ICSCN '07. International Conference on, pages 1–5, 2007.

[KAR03] Mikko Kärkkäinen and Timo Ala-Risku. Automatic identification–applications and technologies. In Logistics Research Network 8th Annual Conference, London UK, 2003.

[KARK01] Mikko Kärkkäinen, Timo Ala-Risku, and Petri Kiianlinna. Item Identification: Applications and Technologies. Helsinki University of Technology, TAI Research Centre, 2001.

[KGB+11] Alexander Kröner, Patrick Gebhard, Boris Brandherm, Benjamin Weyl, Jörg Preißinger, Carsten Magerkurth, and Selcuk Anilmis. Personal shopping support from digital product memories. In PECCS, pages 64–73, 2011.

[KHM+05] Holger Knublauch, Matthew Horridge, Mark A Musen, Alan L Rector, Robert Stevens, Nick Drummond, Phillip W Lord, Natalya Fridman Noy, Julian Seidenberg, and Hai Wang. The protégé owl experience. In OWLED, 2005.

[KhPV+13] Zsolt Kocsi-horváth, Msc Presentation, Ir. H. Vincent, Dr. S. Dulman, and Dr. A. Iosup. Title indoor localization using accidental infrastructure, 2013.

[KR07] C Maria Keet and Mariano Rodrıguez. Toward using bio-ontologies in the semantic web: trade-offs between ontology languages. In Proc. of the AAAI Workshop on Semantic e-Science (SeS 2007), pages 65–68, 2007.

[Lan05] Jeremy Landt. The history of rfid. Potentials, IEEE, 24(4):8–11, 2005.

[LFR+12] Hong Lu, Denise Frauendorfer, Mashfiqui Rabbi, Marianne Schmid Mast, Gokul T. Chittaranjan, Andrew T. Campbell, Daniel Gatica-Perez, and Tanzeem Choudhury. Stresssense: detecting stress in unconstrained acoustic environments using smartphones. In Proceedings of the 2012 ACM Conference on Ubiquitous Computing, UbiComp '12, pages 351–360, New York, NY, USA, 2012. ACM.

[LGO+10] Vitali Loseu, Hassan Ghasemzadeh, Sarah Ostadabbas, Nikhil Raveendranathan, Jacques Malan, and Roozbeh Jafari. Applications of sensing platforms with wearable computers. In Proceedings of the 3rd International Conference on PErvasive Technologies Related to Assistive Environments, page 53. ACM, 2010.

[LSDR10] Antonio Lanata, Enzo Pasquale Scilingo, and Danilo De Rossi. A multimodal transducer for cardiopulmonary activity monitoring in emergency. Information Technology in Biomedicine, IEEE Transactions on, 14(3):817–825, 2010.

[LWW+13] Xiang Liu, Zhibo Wang, Zhi Wang, Shugang Lv, and Tao Guan. A novel real-time traffic information collection system based on smartphone. In Advances in Wireless Sensor Networks, pages 291–303. Springer, 2013.

[Lyn96] N. A. Lynch. Distributed Algorithms. Morgan Kaufmann, San Francisco, CA, 1996.

[Ma11] Zhenqiang Ma. An electronic second skin. Science, 333(6044):830–831, 2011.

[May] Simon Mayer. Communication technologies and their applications beyond communication.

[MESG10] Cambillau Mathieu, El-Shanta Eltaher, Purushotham Sarathy, and Simeu Gilles. Owl ontology for solar uv exposure and human health. Advances in Semantic Computing, Eds. Joshi, Boley & Akerkar, 2:32–51, 2010.

[MPF09] Sabba Maqbool, Henry P Parkman, and Frank K Friedenberg. Wireless capsule motility: comparison of the smartpill® gi monitoring system with scintigraphy for measuring whole gut transit. Digestive diseases and sciences, 54(10):2167–2174, 2009.

[MSM07] Hiroyasu Miwa, S-i Sasahara, and Toshihiro Matsui. Roll-over detection and sleep quality measurement using a wearable sensor. In Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE, pages 1507–1510. IEEE, 2007.

[Mul93] Sape Mullender. Distributed Systems. Addison-Wesley, 2nd edition, 1993.

[NDNG03] Henry Newton-Dunn, Hiroaki Nakano, and James Gibson. Block jam: a tangible interface for interactive music. In Proceedings of the 2003 conference on New interfaces for musical expression, NIME '03, pages 170–177, Singapore, Singapore, 2003. National University of Singapore.

[NP11] Taewoo Nam and Theresa A. Pardo. Conceptualizing smart city with dimensions of technology, people, and institutions. In Proceedings of the 12th Annual International Digital Government Research Conference: Digital Government Innovation in Challenging Times, dg.o '11, pages 282–291, New York, NY, USA, 2011. ACM.

[NR96] Katashi Nagao and Jun Rekimoto. Agent augmented reality: A software agent meets the real world. In Proceedings of the Second International Conference on Multi-Agent Systems (ICMAS-96), pages 228–235, 1996.

[NSS13] Dmitry Namiot and Manfred Sneps-Sneppe. Geofence and network proximity. In Sergey Balandin, Sergey Andreev, and Yevgeni Koucheryavy, editors, Internet of Things, Smart Spaces, and Next Generation Networking, volume 8121 of Lecture Notes in Computer Science, pages 117–127. Springer Berlin Heidelberg, 2013.

[onl] A way forward for glass recycling. Available at `http://www.waste-management-world.com/articles/print/volume-10/issue-1/features/a-way-forward-for-glass-recycling.html` [accessed 09 December 2013,15h41].

[PB08] D. Preuveneers and Y. Berbers. Encoding semantic awareness in resource-constrained devices. Intelligent Systems, IEEE, 23(2):26–33, 2008.

[Per00] J.M. Pereira. Fourth generation: Now, it is personal! In Personal, Indoor and Mobile Radio Communications, 2000. PIMRC 2000. The 11th IEEE International Symposium on, volume 2, pages 1009–1016 vol.2, 2000.

[PJA+10] Shadab A Pathan, Gaurav K Jain, Sohail Akhter, Divya Vohora, Farhan J Ahmad, and Roop K Khar. Insights into the novel three Śd's of epilepsy treatment: drugs, delivery systems and devices. Drug discovery today, 15(17):717–732, 2010.

[PK10] Stevan Preradovic and Nemai Chandra Karmakar. Chipless rfid: bar code of the future. Microwave Magazine, IEEE, 11(7):87–97, 2010.

[RBR+07] Yves Rimet, Yves Brusquet, Dominique Ronayette, Christian Dageville, Marc Lubrano, Eric Mallet, Caroline Rambaud, Christian Terlaud, Jerome Silve, Olivier Lerda, et al. Surveillance of infants at risk of apparent life threatening events (alte) with the bba bootee: a wearable multiparameter monitor. In Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE, pages 4997–5000. IEEE, 2007.

[RFI] Introducing the speedway xarray® reader system. Available at `http://www.impinj.com/`

`Speedway_xArray_RFID_Reader_System.aspx` [accessed 20 February 2014,15h41].

[RMB+98] Mitchel Resnick, Fred Martin, Robert Berg, Rick Borovoy, Vanessa Colella, Kwin Kramer, and Brian Silverman. Digital manipulatives: new toys to think with. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '98, pages 281–287, New York, NY, USA, 1998. ACM Press/Addison-Wesley Publishing Co.

[RN95] Jun Rekimoto and Katashi Nagao. The world through the computer: computer augmented interaction with real world environments. In Proceedings of the 8th annual ACM symposium on User interface and software technology, UIST '95, pages 29–36, New York, NY, USA, 1995. ACM.

[Rob06] C.M. Roberts. Radio frequency identification (rfid). Computers Security, 25(1):18 – 26, 2006.

[RPI04] Hayes Solos Raffle, Amanda J. Parkes, and Hiroshi Ishii. Topobo: a constructive assembly system with kinetic memory. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '04, pages 647–654, New York, NY, USA, 2004. ACM.

[RSMD04] Kay Römer, Thomas Schoch, Friedemann Mattern, and Thomas Dübendorfer. Smart identification frameworks for ubiquitous computing applications. Wireless Networks, 10(6):689–700, 2004.

[SABH09] Roland Spies, Markus Ablaßmeier, Heiner Bubb, and Werner Hamberger. Augmented interaction and visualization in the automotive domain. In Human-Computer Interaction. Ambient, Ubiquitous and Intelligent Interaction, pages 211–220. Springer, 2009.

[Sad07] El Saddik. The potential of haptics technologies. Instrumentation Measurement Magazine, IEEE, 10(1):10–17, Feb 2007.

[SAG+93] Bill N Schilit, Norman Adams, Rich Gold, Michael M Tso, and Roy Want. The parctab mobile computing system. In Workstation Operating Systems, 1993. Proceedings., Fourth Workshop on, pages 34–39. IEEE, 1993.

[Sat01] M. Satyanarayanan. Pervasive computing: vision and challenges. Personal Communications, IEEE, 8(4):10–17, 2001.

[SC12] Arnab Sinha and Paul Couderc. Using owl ontologies for selective waste sorting and recycling. In 9th OWL: Experiences and Directions Workshop (OWLED 2012), Crete, Greece, 2012.

[SC13a] Arnab Sinha and Paul Couderc. A framework for interacting smart objects. In ruSMART 2013 : The 6th conference on Internet of Things and Smart Spaces, Saint Petersburg, Russia, 2013. Springer.

[SC13b] Arnab Sinha and Paul Couderc. Smart bin for incompatible waste items. In Proceedings of the 9th International Conference on Autonomic and Autonomous Systems (ICAS 2013), pages 40–45, Lisbon, Portugal, 2013.

[Sci] The science museum (2004). "urban development". Available at `http://www.makingthemodernworld.org.uk/learning_modules/geography/04.TU.01/?section=2` [accessed 20 November 2013,13h57].

[Sei12] Gildo Seisdedos. Pero¿ qué*% &¡ es una smart cities. Bit, (188):35–37, 2012.

[SGC14] Arnab Sinha, Yann Glouche, and Paul Couderc. Distributed tree structure for composite physical objects. In Proceedings of the 5th International Conference on Ambient Systems, Networks and Technologies (ANT 2014), Hasselt, Belgium, 2014. Elsivier.

[Sha13] Pankaj Sharma. Evolution of mobile wireless communication networks-1g to 5g as well as future prospective of next generation communication network. 2013.

[SIKH82] David Canfield Smith, Charles Irby, Ralph Kimball, and Eric Harslem. The star user interface: an overview. In Proceedings of the June 7-10, 1982, national computer conference, AFIPS '82, pages 515–528, New York, NY, USA, 1982. ACM.

[SK95] Hideyuki Suzuki and Hiroshi Kato. Interaction-level support for collaborative learning: Algoblock Ů an open programming language. In The first international conference on Computer support for collaborative learning, CSCL '95, pages 349–355, Hillsdale, NJ, USA, 1995. L. Erlbaum Associates Inc.

[SK08] M. Schneider and A. Kroner. The smart pizza packing: An application of object memories. In Intelligent Environments, 2008 IET 4th International Conference on, pages 1–8, 2008.

[SMK+10] Peter Stephan, Gerrit Meixner, Holger Koessling, Florian Floerchinger, and Lisa Ollinger. Product-mediated communication through digital object memories in heterogeneous value chains. In Pervasive Computing and Communications (PerCom), 2010 IEEE International Conference on, pages 199–207. IEEE, 2010.

[SSL+07] Lisa K Simone, Nappinnai Sundarrajan, Xun Luo, Yicheng Jia, and Derek G Kamper. A low cost instrumented glove for extended monitoring and functional hand assessment. Journal of neuroscience methods, 160(2):335–348, 2007.

[SSVL+02] Albrecht Schmidt, Martin Strohbach, Kristof Van Laerhoven, Adrian Friday, and Hans-Werner Gellersen. Context acquisition based on load sensing. In UbiComp 2002: Ubiquitous Computing, pages 333–350. Springer, 2002.

[Sto48] Harry Stockman. Communication by means of reflected power. Proceedings of the IRE, 36(10):1196–1204, 1948.

[Sut68] Ivan E. Sutherland. A head-mounted three dimensional display. In Proceedings of the December 9-11, 1968, fall joint computer conference, part I, AFIPS '68 (Fall, part I), pages 757–764, New York, NY, USA, 1968. ACM.

[TA] INRIA Team ACES. Bin that thinks @ONLINE. Available at `http://binthatthink.inria.fr/` [accessed 04 January 2014, 16h08].

[Thi] Thingful aims to be the google of the internet of things. Available at `http://www.telecoms.com/206211/thingul-aims-to-be-the-google-of-the-internet-of-things/` [accessed 20 February 2014,15h45].

[UI99] Brygg Ullmer and Hiroshi Ishii. mediablocks: tangible interfaces for online media. In CHI '99 Extended Abstracts on Human Factors in Computing Systems, CHI EA '99, pages 31–32, New York, NY, USA, 1999. ACM.

[VBC12] J Verdonck, Michel Banâtre, and Paul Couderc. Ubi-post briefcase: a content-oriented security solution for high value shipping. In Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on, pages 492–494. IEEE, 2012.

[VPS+10] Athanasios S. Voulodimos, Charalampos Z. Patrikakis, Alexander B. Sideridis, Vasileios A. Ntafis, and Eftychia M. Xylouri. A complete farm management system based on animal identification using {RFID} technology. Computers and Electronics in Agriculture, 70(2):380 – 388, 2010. <ce:title>Special issue on Information and Communication Technologies in Bio and Earth Sciences</ce:title>.

[(W3a] World Wide Web Consortium (W3C). Owl web ontology language overview @ONLINE. Available at `http://www.w3.org/TR/owl-features/` [accessed 04 January 2014, 16h08].

[(W3b] World Wide Web Consortium (W3C). Owl web ontology language reference @ONLINE. Available at `http://www.w3.org/TR/owl-ref/` [accessed 04 January 2014, 16h08].

[Wan06] Roy Want. An introduction to rfid technology. Pervasive Computing, IEEE, 5(1):25–33, 2006.

[WCH+09] Yu-Chi Wu, Pei-Fan Chen, Zhi-Huang Hu, Chao-Hsu Chang, Gwo-Chuan Lee, and Wen-Ching Yu. A mobile health monitoring system using rfid ring-type pulse sensor. In Dependable, Autonomic and Secure Computing, 2009. DASC'09. Eighth IEEE International Conference on, pages 317–322. IEEE, 2009.

[Wei91] Mark Weiser. The computer for the 21st century. Scientific American, 265(3):94–104, September 1991. http://www.ubiq.com/hypertext/weiser/SciAmDraft3.html.

[Wei93] Mark Weiser. Some computer science issues in ubiquitous computing. Communications of the ACM, 36(7):75–84, 1993.

[Wei03] Stephen August Weis. Security and privacy in radio-frequency identification devices. PhD thesis, Massachusetts Institute of Technology, 2003.

[WHFG92] Roy Want, Andy Hopper, Veronica Falcão, and Jonathan Gibbons. The active badge location system. ACM Transactions on Information Systems (TOIS), 10(1):91–102, 1992.

[wik] Radio-frequency identification. Available at `http://en.wikipedia.org/wiki/Radio-frequency_identification` [accessed 10 December 2013,10h24].

[WP02] Peta Wyeth and Helen C. Purchase. Tangible programming elements for young children. In CHI '02 Extended Abstracts on Human Factors in Computing Systems, CHI EA '02, pages 774–775, New York, NY, USA, 2002. ACM.

[WS09] Doug Washburn and Usman Sindhu. Helping cios understand Şsmart cityŤ initiatives. Growth, 2009.

[YL09] Bo Yan and Danyu Lee. Application of rfid in cold chain temperature monitoring system. In Computing, Communication, Control, and Management, 2009. CCCM 2009. ISECS International Colloquium on, volume 2, pages 258–261, 2009.

[YLC+13] Chuang-Wen You, Nicholas D Lane, Fanglin Chen, Rui Wang, Zhenyu Chen, Thomas J Bao, Martha Montes-de Oca, Yuting Cheng, Mu Lin, Lorenzo Torresani, et al. Carsafe app: Alerting drowsy and distracted drivers using dual cameras on smartphones. submitted to ACM Mobisys, 2013.

[YMdOB+12] Chuang-Wen You, Martha Montes-de Oca, Thomas J Bao, Nicholas D Lane, Hong Lu, Giuseppe Cardone, Lorenzo Torresani, and Andrew T Campbell. Carsafe: a driver safety app that detects dangerous driving behavior using dual-cameras on smartphones. In UbiComp, pages 671–672, 2012.

[YT12] Koji Yatani and Khai N Truong. Bodyscope: a wearable acoustic sensor for activity recognition. In Proceedings of the 2012 ACM Conference on Ubiquitous Computing, pages 341–350. ACM, 2012.

[ZGO13] Oren Zuckerman and Ayelet Gal-Oz. To {TUI} or not to tui: Evaluating performance and preference in tangible vs. graphical user interfaces. International Journal of Human-Computer Studies, 71(7Ű8):803 – 820, 2013.

[ZR03] Oren Zuckerman and Mitchel Resnick. System blocks: A physical interface for system dynamics simulation. In In Proceedings of CHI Š03, ACM, pages 810–811. Press, 2003.

List of Figures

1.1 Evolution of Mobile and Pervasive Computing from Distributed Systems [Sat01] 9
1.2 General Model - Pervasive Computing 11
1.3 Smart trolley with interactive features 11
1.4 A comparison of HCI styles [RN95] 15
1.5 Smartphone sensors 18
1.6 Sensor Networks in battlefield 19
1.7 (a) Infrastructured Wireless LAN; (b) Ad-hoc Wireless LAN. 23
1.8 Automatic doors using Infrared 26
1.9 Interaction spaces using infrared beams 26
1.10 Block diagram of a typical RFID system [PK10] 29
1.11 (a) RFID tag form factors (b) Inside RFID tag 31
1.12 Using two RFID readers for “spatial” coverage 34
1.13 Long and short range communication 36
1.14 EPC Tag Format [Rob06] 37
1.15 Data processing architectures 39

2.1 Smart Coffee Maker with and without using Network 44
2.2 E-balance using RFID reader 46
2.3 Tangible Data for food items 47
2.4 Tangible Data for Box 48
2.5 Representation of recyclable material percentage as Tangible Data 50
2.6 Representation of recyclable material percentage as Tangible Data 51
2.7 Commonly used Architecture for Systems 52
2.8 Conditions to describe a category 53
2.9 InoT formed 54
2.10 Bookshelf with related objects (books) 55
2.11 Complex Object 56
2.12 Composite representation of a chair as hierarchy 57
2.13 Coupled Object using Complete Graph 58
2.14 Coupled Object using 2-Regular Graph 59
2.15 UbInspect using (a) 3-Regular Graph (b) 4-Regular Graph [GC12] 59
2.16 Integrity check fails: (a) 4 missing (b) 5 missing Objects [GC12] 60

2.17 Neighbours of Node 1 in (a) Fully Connected (b) 3-Regular (c) Vertex-labeled Graph . . . 61
2.18 (a) Toolbox as clique (b) Adding tools to different compartments [SC13a] . . . 63
2.19 Smart Tool Box Application (a) Initial grouping of Tool Box and contents (b) Integrity Verification of the Tool Box (c) Acknowledgement on adding tool numbered 26 (d) Notification for tool(s) yet missing . . . 65
2.20 Encoding performance . . . 66
2.21 Sample tree structure to be distributed among leaf nodes . . . 68
2.22 Memory space organization of the leaves . . . 69
2.23 Tree of depth two . . . 70
2.24 Injective function f on a tree of depth two . . . 70
2.25 Tree of depth n . . . 70
2.26 Tree of depth $n+1$ . . . 71
2.27 Elementary trees marked in coloured circles . . . 72
2.28 Elementary trees added to end of List L . . . 75
2.29 Tree from List L added under Elementary tree and then inserted into List . . . 75
2.30 Adding sibling leaf nodes . . . 76
2.31 Comparing structural composition of nodes . . . 76
2.32 (a) Memory usage of leaves in complete trees; (b) Increasing depth consumes less memory in leaves. . . . 78

3.1 Smart Bin Infrastructure as a Sensor Network . . . 81
3.2 Waste flow and global architecture of the system [GC13] . . . 83
3.3 A collective bin using the RFID technology. . . . 85
3.4 Prototype demonstration: (1) Kitchen bin (2) Collective bin showing status (3) Truck's Application Screen (4) Operator's status application at city level . . . 87

4.1 Sectorwise inventorying of Smart Library using beam-forming RFID antenna . 90

A.1 Object properties as mapped in OWL . . . 96
A.2 Data property $\mathrm{hasStatus}$ . . . 96
A.3 Sample Ontology with Classes and Conditions . . . 97
A.4 Describing Categories . . . 97
A.5 External Conditions . . . 98
A.6 Dashboard . . . 98
A.7 SWRL Rules . . . 99
A.8 Smart Bin . . . 100
A.9 Hazard Detection Application for Waste . . . 100

C.1 Elements d'un système d'informatique diffuse . . . 107
C.2 Communications globales et communications à courte portée . . . 108
C.3 Architectures de traitement de données . . . 110
C.4 Bibliothèque d'objets reliés (livres) . . . 112
C.5 Objets liés exploitant un graphe complet . . . 112
C.6 Représentation composite d'une chaise sous la forme d'une hiérarchie . . . 113

C.7 Inventaire sectoriel d'une bibliothèque avec un lecteur RFID à antenne intelligente 115

Self-Describing Objects with Tangible Data Structures

Pervasive computing or ambient computing aims to integrate information systems into the environment, in a manner as transparent as possible to the users. It allows the information systems to be tightly coupled with the physical activities within the environment. Everyday used objects, along with their environment, are made smarter with the use of embedded computing, sensors etc. and also have the ability to communicate among themselves.

In pervasive computing, it is necessary to sense the real physical world and to perceive its "context"; a high level representation of the physical situation. There are various ways to derive the context. Typically, the approach is a multi-step process which begins with sensing. Various sensing technologies are used to capture low level information of the physical activities, which are then aggregated, analyzed and computed elsewhere in the information systems, to become aware of the context. Deployed applications then react, depending on the context situation. Among sensors, RFID is an important emerging technology which allows a direct digital link between information systems and physical objects. Besides storing identification data, RFID also provides a general purpose storage space on objects, enabling new architectures for pervasive computing.

In this thesis, we defend an original approach adopting the later use of RFID i.e. a digital memory integrated to real objects. The approach uses the principle where the objects self-support information systems. This way of integration reduces the need of communication for remote processing. The principle is realized in two ways. First, objects are piggybacked with semantic information, related to itself; as self-describing objects. Hence, relevant information associated with the physical entities are readily available locally for processing. Second, group of related objects are digitally linked using dedicated or ad-hoc data structure, distributed over the objects. Hence, it would allow direct data processing - like validating some property involving the objects in proximity. This property of physical relation among objects can be interpreted digitally from the data structure; this justifies the appellation "Tangible Data Structures".

Unlike the conventional method of using identifiers, our approach has arguments on its benefits in terms of privacy, scalability, autonomy and reduced dependency with respect to infrastructure. But its challenge lies in the expressivity due to limited memory space available in the tags.

The principles are validated by prototyping in two different application domains. The first application is developed for waste management domain that helps in efficient sorting and better recycling. And the second, provides added services like assistance while assembling and verification for composite objects, using the distributed data structure across the individual pieces.

Objets auto-descriptifs avec des structures de données tangibles

L'informatique ubiquitaire, ou informatique omniprésente, a pour but l'intégration de systèmes d'information dans l'environnement de façon aussi transparente que possible pour l'utilisateur. Elle permet aux systèmes d'information d'être directement couplés aux activités physiques ayant lieu dans l'environnement. Les objets communs, ainsi que leur environnement, sont rendus intelligents grâce à des sytèmes embarqués, des capteurs, etc. Ils acquièrent aussi la capacité de communiquer entre eux.

En informatique ubiquitaire, l'observation du monde physique et de son "contexte" (une représentation haut niveau de la situation physique) est essentielle. Il existe de nombreux moyens pour observer le contexte. Typiquement, cela consiste en un traitement en plusieurs étapes commençant par la récupération de données brutes issues de capteurs. Diverses technologies de capteurs sont utilisées pour la récupération d'informations de bas niveau sur les activités physiques en cours. Ces données sont ensuite rassemblées, analysées et traitées ailleurs dans les systèmes d'information afin d'offrir une reconnaissance de contexte. Les applications déployées réagissent alors en fonction du contexte/de la situation détecté(e). Parmis les capteurs utilisés, les tags RFID, une technologie émergente, permettent de créer un lien virtuel direct entre les objets physiques et les systèmes d'information. En plus de stocker des identifiants, ils offrent un espace mémoire générique aux objets auxquels ils sont attachés, offrant de nouvelles possibilités d'architectures en informatique omniprésente.

Dans cette thèse, nous proposons une approche originale tirant parti de l'espace mémoire offerts aux objets réels par les tags RFID. Dans notre approche, les objets supportent directement le système d'information. Ce type d'intégration permet de réduire les communications requises par le traitement à distance. Pour ce faire, des données sémantiques sont tout d'abord attachées aux objets afin de les rendre auto-descriptifs. Ainsi, les données pertinentes concernant une entité physique sont directement disponibles pour un traitement local. Les objets peuvent ensuite être liés virtuellement grâce à des structures de données dédiées ou ad hoc et distribuées sur les objets eux-mêmes. Ce faisant, le traitement des données peut se faire de façon directe. Par exemple, certaines propriétés peuvent être vérifiées localement sur un ensemble d'objets. Une relation physique peut être déduite directement de la structure de données, d'où le nom de "structures de données tangibles". Vis-à-vis des approches conventionnelles tirant parti des identifiants, notre approche offrent des avantages en termes de vie privée, de mise à l'échelle, d'autonomie et d'indépendance vis-à-vis des infrastructures. Le défi se situe au niveau de son expressivité limitée à cause du faible espace mémoire disponible sur les tags RFID.

Les principes sont validés dans deux prototypes aux applications différentes. Le premier prototype est développé dans le domaine de la gestion de déchets afin d'aider le tri et d'améliorer le recyclage. Le deuxième offre des services supplémentaires, tels qu'une assistance lors du montage et de la vérification d'objets composés de plusieurs parties, grâce aux structures de données distribuées sur les différentes parties.