



UNIVERSIDADE NOVA DE LISBOA  
Faculdade de Ciências e Tecnologia  
Departamento de Informática

---

# Uma Infra-estrutura Computacional para Aumentar Objectos Reais com Informação Adicional

Por

Rui Miguel Neves G Madeira

Dissertação apresentada na Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa para a obtenção  
do grau de Mestre em Engenharia Informática

**Orientador:** Professor Doutor Nuno Manuel Robalo Correia

Lisboa

2007







## Resumo

A Computação Ubíqua tem como objectivo principal a construção de sistemas computacionais que suportem e facilitem o quotidiano dos utilizadores, mas sendo o menos intrusivos possível. Idealmente, estes nem se aperceberiam desse seu papel, pois toda a estrutura do sistema estaria de tal forma integrada no ambiente que se tornaria invisível, fazendo desaparecer a consciência de que se está perante um sistema computacional. Nos últimos anos, tem-se intensificado a investigação nesta área e são muitos os avanços tecnológicos, mas o cenário actual ainda se encontra longe de um quotidiano preenchido por sistemas ubíquos invisíveis.

Esta dissertação segue uma perspectiva que defende uma aproximação prudente e iterativa à visão de Computação Ubíqua. Neste âmbito, é estudada e apresentada uma proposta de arquitectura e sistema, baseada em tecnologias reconhecidas, com um objectivo aplicacional restrito. Este sistema aumenta os objectos físicos presentes num ambiente através da apresentação de informação multimédia adicional. A informação que aumenta um objecto é contextual, baseando-se não só no objecto, mas também na sua localização específica, no espaço e no perfil dos utilizadores.

Foi desenhada uma arquitectura genérica para a implementação de infra-estruturas computacionais deste tipo em qualquer espaço físico. Esta tem em consideração a distribuição do sistema pelas várias entidades presentes no ambiente, que são utilizadores, dispositivos e objectos. A apresentação da informação é feita onde e quando é necessária, sendo o mais discreta possível, e com interacção passiva. Numa abordagem complementar, permite-se uma interacção activa, desde que solicitada pelo utilizador.

Foram desenvolvidos dois protótipos da infra-estrutura, cada um com base numa tecnologia madura para detecção e comunicação - RFID e Bluetooth. Estes protótipos permitiram que estas tecnologias fossem testadas, servindo também como prova de viabilidade da infra-estrutura.

**Palavras Chave:** computação ubíqua, objectos aumentados, informação contextual, RFID, Bluetooth, dispositivos móveis, ecrãs públicos.



## Abstract

Ubiquitous Computing has the main goal of building computer systems that support and facilitate the daily lives of users, but being the least intrusive possible. Ideally, these wouldn't even be aware of their role, because the whole structure of the system would be so integrated in the environment that it would become invisible. In recent years, the research in this area has intensified and there are many technological advances, but the current scenario is still far away from an everyday life fulfilled with invisible ubiquitous systems.

This dissertation follows a perspective that advocates a cautious and iterative approach to the vision of Ubiquitous Computing. In this context, it is studied and presented a proposal of an architecture and a system, based on recognized technologies, with a restricted application goal. This system augments the physical objects present in an environment with additional multimedia information. The information that augments an object is contextual, based on not only in the object, but also in its specific location, in the physical space and in the profile of the users.

A generic architecture was designed to implement this kind of computing infrastructure in any physical space. It takes into account the distribution of the system by the various entities in the environment, which are users, devices and objects. The presentation of the information is made where and when is needed, being as unobtrusive as possible, and with passive interaction. In a complementary approach, it allows an active interaction, as long as requested by the user.

Two prototypes of the infrastructure have been developed, each one is based on a mature technology for detection and communication - RFID e Bluetooth. These prototypes allowed that these technologies were tested, also serving as proof of the infrastructure's viability.

**Keywords:** ubiquitous computing, augmented objects, contextual information, RFID, Bluetooth, mobile devices, public displays.



## **Agradecimentos**

É chegado o momento de pensar em todos aqueles que contribuíram com algo de precioso para o caminho que me trouxe até este ponto da Vida.

O primeiro agradecimento é dirigido ao arquitecto desta construção, Prof. Nuno Correia, pelas demonstrações de dedicação, incentivo e amizade. Não posso esquecer a confiança que continuamente depositou em mim, desde a oferta do tema para dissertação até à sua conclusão.

Agradeço de forma especial à minha esposa Poli e aos meus pais, Maria Deolinda e Francisco, pois são eles os grandes baluartes nesta minha construção. Agradeço-lhes a confiança, o apoio incondicional, o carinho e o Amor que sempre demonstraram ao longo de todos os anos. À Poli, tenho ainda que agradecer uma maior paciência evidenciada em períodos mais delicados. Foi determinante.

Aos restantes membros da minha família mais próxima agradeço o carinho e a confiança que sempre me ofereceram ao longo da minha vida. A título de exemplo, refiro apenas as dádivas sentimentais que sempre recebi dos meus Avós. São incontáveis as saudades!

Ao Tobias, meu fiel e dedicado companheiro, agradeço as infindáveis horas que dedicou ao acompanhamento desta dissertação.

Agradeço aos meus colegas Anacleto Correia e Patrícia Macedo pelas dicas e conversas que foram essenciais para que algumas arestas fossem limadas com sucesso. Aos meus ex-alunos Gilberto Catarino, Andreia Costa, Luís Moita, Hugo Branco e Pedro Soeiro, agradeço pelas discussões de ideias e pelo trabalho desenvolvido que muito ajudou a sustentar a tese apresentada.

Por fim, agradeço, na pessoa do Bruno Silva, a todos os meus amigos e colegas da Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa e da Escola Superior de Tecnologia de Setúbal, que estiveram ao meu lado incentivando-me e apoiando-me para que o árduo caminho fosse ultrapassado de forma um pouco mais suave.



# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Enquadramento . . . . .	1
1.1.1	Motivação . . . . .	1
1.1.2	Objectivos . . . . .	3
1.2	Metodologia . . . . .	5
1.3	Mapa da Tese . . . . .	6
<b>2</b>	<b>Enquadramento Teórico</b>	<b>7</b>
2.1	Uma Visão . . . . .	7
2.2	Introdução à Computação Ubíqua . . . . .	8
2.2.1	Questões Pertinentes em Computação Ubíqua . . . . .	11
2.3	Elementos-Chave num Ambiente de Computação Ubíqua . . . . .	12
2.4	Informação Contextual . . . . .	15
2.5	Projectos e Sistemas de Computação Ubíqua . . . . .	18
<b>3</b>	<b>Trabalho Relacionado</b>	<b>23</b>
3.1	A Computação Embutida e as Tecnologias . . . . .	23
3.2	Comunicação <i>Wireless</i> . . . . .	24
3.2.1	Infravermelhos . . . . .	24
3.2.2	Radiofrequência . . . . .	25
3.2.2.1	Wi-Fi . . . . .	26
3.2.2.2	Bluetooth . . . . .	27
3.2.2.3	Comparação das Normas IEEE 802 . . . . .	29
3.2.2.4	NFC ( <i>Near Field Communication</i> ) . . . . .	30
3.3	Identificação e Localização . . . . .	31
3.3.1	Identificadores de Endereço Único . . . . .	32

3.3.2	<i>Visual Tags</i> . . . . .	32
3.3.3	RFID ( <i>Radio Frequency Identification</i> ) . . . . .	33
3.3.4	GPS ( <i>Global Positioning System</i> ) . . . . .	36
3.3.5	RTLS ( <i>Real Time Locating Systems</i> ) . . . . .	37
3.4	Sensores . . . . .	39
3.4.1	Redes de Sensores . . . . .	40
3.5	Plataformas de Desenvolvimento . . . . .	41
3.6	Dispositivos Móveis Pessoais . . . . .	43
3.7	Ecrãs . . . . .	44
3.8	Projectos e Sistemas Relacionados . . . . .	46
<b>4</b>	<b>Princípios e Arquitectura</b> . . . . .	<b>57</b>
4.1	Princípios . . . . .	57
4.1.1	Sistema de Informação Ambiental . . . . .	58
4.1.2	Interacções Acidentais e Co-Presença . . . . .	59
4.1.2.1	Noção do Espaço (Ambiente) . . . . .	60
4.1.3	Ecrãs Situados . . . . .	61
4.1.4	Informação Contextual Embutida . . . . .	64
4.1.5	Ubiquidade dos Dispositivos Móveis . . . . .	65
4.2	Arquitectura da infra-estrutura . . . . .	66
4.2.1	Requisitos . . . . .	66
4.2.2	Arquitectura Genérica . . . . .	67
4.2.2.1	Os Módulos Centrais da Arquitectura . . . . .	70
4.2.2.2	Os Módulos para Aumentação Móvel . . . . .	76
<b>5</b>	<b>Realização do Protótipo divingForPearls</b> . . . . .	<b>79</b>
5.1	Tipo de Cenário e Actividades . . . . .	79
5.2	Enquadramento Organizacional . . . . .	82
5.3	Análise de Requisitos . . . . .	83
5.4	Instanciação da Arquitectura . . . . .	84
5.4.1	Componente de Rastreo . . . . .	85
5.4.1.1	Dispositivo de Rastreo . . . . .	85
5.4.1.2	Middleware . . . . .	88
5.4.1.3	Gestor de Rastreo . . . . .	90

5.4.2	Componente Aplicacional . . . . .	91
5.4.2.1	Módulo Personalização . . . . .	93
5.4.2.2	Módulo Conteúdos de Objectos . . . . .	93
5.4.2.3	Interpretador Contextual . . . . .	94
5.4.2.4	Gestor de Composição . . . . .	95
5.4.2.5	Servidor de Informação . . . . .	96
5.4.3	Componente Interface Ecrã . . . . .	97
5.4.3.1	Dispositivo de Visualização . . . . .	98
5.4.3.2	Gestor de Dispositivo . . . . .	98
5.4.3.3	Apresentação e Composição de Apresentação . . . . .	98
5.4.4	Configuração Final . . . . .	100
5.5	Aplicação para Gestão de Dados . . . . .	101
<b>6</b>	<b>Realização do Protótipo directInfo2u</b>	<b>105</b>
6.1	Tipo de Cenário e Actividades . . . . .	105
6.2	Enquadramento Organizacional . . . . .	107
6.3	Análise de Requisitos . . . . .	108
6.4	Instanciação da Arquitectura . . . . .	109
6.4.1	Componente de Rastreo . . . . .	110
6.4.1.1	Dispositivo de Rastreo . . . . .	111
6.4.1.2	Middleware . . . . .	112
6.4.1.3	Gestor de Rastreo . . . . .	114
6.4.2	Componente Aplicacional . . . . .	117
6.4.2.1	Módulos Personalização e Conteúdos de Objectos . . . . .	117
6.4.2.2	Gestor e Fila de Eventos . . . . .	118
6.4.2.3	Interpretador Contextual . . . . .	119
6.4.2.4	Gestor de Composição . . . . .	120
6.4.2.5	Gestor de Pedidos . . . . .	121
6.4.2.6	Servidor de Informação . . . . .	121
6.4.3	Componente de Interface Ecrã . . . . .	121
6.4.3.1	Dispositivo de Visualização e Gestor de Dispositivo . . . . .	122
6.4.3.2	Apresentação e Composição de Apresentação . . . . .	122
6.4.4	Componente de Rastreo - Módulo ICCA . . . . .	123

6.4.5	Módulo Aumentação Móvel . . . . .	124
6.4.5.1	Tecnologia de Comunicação . . . . .	125
6.4.5.2	Gestor de Comunicação . . . . .	125
6.4.5.3	Gestor Móvel e Repositório RMS . . . . .	125
6.4.5.4	Apresentação . . . . .	127
6.4.6	Configuração Final . . . . .	128
<b>7</b>	<b>Conclusões e Trabalho Futuro</b>	<b>129</b>
7.1	Resultados Obtidos . . . . .	130
7.2	Contribuições . . . . .	136
7.3	Trabalho Futuro . . . . .	136
	<b>Bibliografia</b>	<b>139</b>
	<b>Apêndices</b>	<b>155</b>
<b>A</b>	<b>Plataforma RFID Anywhere</b>	<b>155</b>
A.1	Componentes da Plataforma <i>Middleware</i> . . . . .	155
A.2	Simulação Realizada para o Protótipo divingForPearls . . . . .	159

# Lista de Figuras

2.1	(esquerda) As três ondas da computação [1]; (direita) ParcTab: Um sistema de computação <i>context-aware</i> [2]. . . . .	8
2.2	(esquerda) A nova dimensão no mundo das ICTs; (direita) Miniaturização em direcção à <i>Internet of Things</i> [3]. . . . .	10
2.3	Um modelo para Informação Contextual [4]. . . . .	17
2.4	4 gerações do Active Badge [5]: (esquerda em baixo) Primeira versão (código de 5 bits), (direita em baixo) segunda versão (10 bits), (esquerda em cima) terceira versão e (direita em cima) quarta e actual versão (48 bits, bi-direccional e com microprocessador 87C751). . . . .	19
2.5	(esquerda) ParcPad, (centro) Liveboard e (direita) ParcTab da Xerox's PARC [1].	20
3.1	Comunicação Bluetooth através do adaptador USB. . . . .	27
3.2	Bluetooth: Arquitectura completa da pilha de protocolos [6]. . . . .	28
3.3	Quadro comparativo de protocolos de radiofrequência [6]. . . . .	29
3.4	Exemplos de <i>visual tags</i> da CyberCode [7]. . . . .	32
3.5	(esquerda) ETH VisualCodes: Telemóvel a fotografar uma <i>visual code</i> [8]; (direita) ARToolkit: Visualização de elemento virtual sobre uma <i>visual tag</i> [9]. . . . .	33
3.6	Infra-estrutura típica de um sistema RFID. . . . .	34
3.7	RFID: (cima) Antenas e (baixo) leitores (fixo e móvel). . . . .	35
3.8	<i>Tags</i> para RTLS: (esquerda) UWB, (centro) Wi-Fi e (direita) RFID Activo. . . . .	38
3.9	(esquerda) BlueBoard e (direita) GroupCast: Em contexto de utilização. . . . .	46
3.10	(esquerda) Hermes Photo Display e (direita) Mobilenin: Apresentação de informação no ecrã. . . . .	47
3.11	(esquerda) BluScreen e (direita) iRemote: Em contexto de utilização. . . . .	48
3.12	magicmirror: Funcionalidades disponibilizadas ao utilizador. . . . .	49

3.13 eCampus: (esquerda) Um ecrã grande e (direita) aplicação iCapture num dispositivo móvel. . . . .	52
3.14 Sistema AWARE: Em contexto de utilização. . . . .	52
3.15 Knight’s Castle: (esquerda) Antenas RFID embutidas no <i>playset</i> e (direita) <i>tags</i> RFID colocadas em peças. . . . .	53
3.16 Rememberer: (esquerda) Equipamento e (direita) exemplo de uma página Web gerada. . . . .	55
3.17 ActivitySpot: Em contexto de utilização. . . . .	56
4.1 Cenário Genérico: Detecção (do utilizador), aumentação visual (do objecto) e interacção activa (entre utilizador e objecto) [adaptado de [10]]. . . . .	58
4.2 Arquitectura Genérica da Infra-estrutura. . . . .	69
4.3 Módulos da Arquitectura Genérica (Objecto Aumentado). . . . .	71
4.4 Arquitectura do Interpretador Contextual. . . . .	75
4.5 Módulos adicionais da arquitectura: (esquerda) Módulo do dispositivo móvel e (direita) adição do ICCA ao módulo Rastreo. . . . .	76
5.1 divingForPearls: Arquitectura típica de um sistema RFID [adaptado de [11]]. . . . .	84
5.2 divingForPearls - Dispositivo de Rastreo: (esquerda) Leitor Alien e (direita) antena RFID utilizada. . . . .	86
5.3 divingForPearls - Dispositivo de Rastreo: (esquerda) <i>Badge</i> com <i>tag</i> adesiva e (direita) exemplo de <i>tag</i> utilizada. . . . .	87
5.4 divingForPearls - RFID Anywhere: Relações entre os seus Componentes. . . . .	89
5.5 divingForPearls - Gestor de Rastreo (BM divingForPearls): Os 4 estados considerados pelo BM para a actividade de <i>tags</i> . . . . .	91
5.6 divingForPearls - Gestor de Rastreo: XML de uma detecção criado pelo BM. . . . .	91
5.7 divingForPearls - Interpretador Contextual: Documentos XML passados para o Gestor de Composição. . . . .	95
5.8 divingForPearls - Gestor de Composição: Um documento XML para o ecrã 1. . . . .	96
5.9 divingForPearls - Apresentação: Visualização de informação adicional sobre uma peça. . . . .	100
5.10 divingForPearls: Proposta de configuração lógica e física. . . . .	101
5.11 divingForPearls - Gestão de Dados: Ecrã ”Servidor”. . . . .	101

5.12	divingForPearls - Gestão de Dados: (esquerda) Ligação à consola de administração do RFID Anywhere e (direita) introdução da porta TCP para escolher BM. . . . .	102
5.13	divingForPearls - Gestão de Dados: Ecrã "Clientes". . . . .	102
5.14	divingForPearls - Gestão de Dados: Ecrãs (esquerda) "Estado" e (direita) "Simular Objecto". . . . .	103
6.1	directInfo2u: Arquitectura detalhada de uma aplicação Java para Bluetooth [12].	110
6.2	directInfo2u: Arquitectura Bluetooth/Java implementada para o servidor. . . . .	111
6.3	directInfo2u - Dispositivo de Rastreo: O adaptador USB de Bluetooth utilizado.	112
6.4	directInfo2u - Middleware: Tabela comparativa com as implementações da API JSR-82 para J2SE. . . . .	113
6.5	directInfo2u - Gestor de Rastreo: Pacote JSR-82 para procurar dispositivos [13].	114
6.6	directInfo2u - Gestor de Rastreo: Mensagem XML de uma detecção. . . . .	116
6.7	directInfo2u - Gestor de Composição: Um exemplo de documento XML para ser lido por um dispositivo móvel. . . . .	120
6.8	directInfo2u - Apresentação: Applet com informação genérica para utilizador registado e com indicação de como registar para um outro. . . . .	122
6.9	directInfo2u - Apresentação: (esquerda) Applet com informação genérica para dispositivo não registado e (direita) Applet para utilizador registado. . . . .	123
6.10	directInfo2u: Arquitectura Bluetooth/Java implementada para o dispositivo móvel.	124
6.11	directInfo2u: Arquitectura CLDC/MIDP com integração de Bluetooth [14]. . . . .	124
6.12	directInfo2u - Aplicação Móvel: (esquerda) O menu para acesso a informação privada e (direita) leitura de informação recebida (simulado no J2ME Wireless Toolkit). . . . .	127
6.13	directInfo2u - Aplicação Móvel: Um outro exemplo em que o utilizador é actualizado com notícias sobre dois itens (visualizado num HTC TyTn). . . . .	127
6.14	directInfo2u: Arquitectura de configuração física final. . . . .	128
7.1	Quadro comparativo de sistemas e projectos relacionados. . . . .	133
A.1	RFID Anywhere: Gestor de serviços da consola de administração com serviços criados para o protótipo. . . . .	156
A.2	RFID Anywhere: Editor de dados do RFID Simulator. . . . .	157

A.3 divingForPearls - RFID Anywhere: Exemplo de um relatório XML gerado de um <i>Report Engine MP</i> . . . . .	160
---	-----

# Lista de Tabelas

3.1	IEEE 802.11: Características dos seus protocolos para transferência de dados. . .	27
5.1	divingForPearls - Dispositivo de Rastreo: Principais características do leitor RFID ALR-8800 da Alien. . . . .	87
5.2	divingForPearls - Personalização: Perfis criados. . . . .	93
5.3	divingForPearls - Composição de Apresentação: Valores definidos para TipoInfo.	99
6.1	directInfo2u - Personalização: Perfis criados no módulo. . . . .	118
6.2	directInfo2u: <i>Parsers</i> XML para a plataforma MIDP. . . . .	126



# Capítulo 1

## Introdução

O presente capítulo tem como objectivo introduzir a dissertação ao leitor. Faz-se um enquadramento no qual se apresenta a motivação e se enumeram os objectivos a alcançar com o trabalho realizado. Descreve-se a metodologia seguida na realização do trabalho. Por último, apresenta-se o mapa da tese, onde é descrita a estrutura deste documento.

### 1.1 Enquadramento

O trabalho apresentado nesta tese insere-se num ambiente de investigação de novas tecnologias e de novos paradigmas de computação, enquadrando-se fortemente na Computação Ubíqua e tendo uma componente de Computação Multimédia. Insere-se em projectos de investigação desenvolvidos no âmbito do Grupo de Multimédia Interactiva<sup>1</sup> (*Interactive Multimedia Group* [IMG]), integrado no Centro de Informática e Tecnologias de Informação (CITI) do Departamento de Informática da Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa (DI/FCT/UNL). O grupo desenvolve trabalhos nas áreas de multimédia, hipermédia, realidade aumentada e mista, personalização, geração automática de informação, e mais recentemente, Computação Ubíqua. Espera-se que os resultados obtidos com o presente trabalho sejam aplicados no projecto financiado pela FCT (Fundação para a Ciência e a Tecnologia) com a referência PTDC/CCI/68595/2006.

#### 1.1.1 Motivação

*"The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it"*. Mark Weiser iniciou deste modo o

---

<sup>1</sup><http://img.di.fct.unl.pt>

embrionário artigo de 1991 [15] no qual delineava a sua visão para computação ubíqua. A essência da sua visão baseava-se na criação de ambientes saturados de computação e comunicação, mas integradas de forma suave e discreta no ambiente envolvente. Para ele, elementos especializados de *hardware* e de *software* serão tão ubíquos que ninguém dará pela sua presença. Quando articulada, esta era uma visão muito distante da sua época e, até hoje, poucos foram os resultados realmente palpáveis que surgiram a partir dela.

Entretanto, a realização técnica desta visão tornou-se possível, muito devido aos recentes avanços nas tecnologias de miniaturização, de conectividade *wireless* e de computação embutida<sup>2</sup>. Estas permitem a integração de diversas capacidades computacionais nos mais diversos objectos e dispositivos do quotidiano. Por outro lado, assumindo que as pessoas têm acesso a PCs (*Personal Computers*) em casa ou no emprego, colocar um PC em cada carro e um *smart phone* em cada bolso, o que não estará longe de acontecer, será uma simples infra-estrutura que pode funcionar enquanto aproximação a uma construção estruturada de um mundo computacional segundo a visão de Weiser. Ainda para mais, os artefactos (e.g., ecrãs públicos, telemóveis, tags RFID, GPS) que são já considerados úteis para a computação ubíqua estão a ficar cada vez mais baratos, possibilitando uma abrangente, vulgar e natural disponibilidade nos espaços públicos.

A multiplicidade e a diversidade de entidades computacionais que fazem parte dos sistemas de computação ubíqua implicam um certo número de desafios técnicos. Por um lado, as diferenças em factores como a forma, a funcionalidade, e as capacidades técnicas resulta numa grande diversidade de *software* e *hardware*, e numa heterogeneidade quanto a capacidades finais. Por outro lado, os sistemas são geralmente descentralizados, com uma grande volatilidade quanto a relações cooperativas e a topologias, resultantes da mobilidade, da interacção espontânea, e da comunicação *wireless ad hoc*. Ainda por cima, uma característica particular da computação ubíqua é centrar-se no utilizador. Estes novos sistemas devem ser explicitamente projectados para suportar as pessoas em situações do seu dia-a-dia. Estas são apenas as questões mais imediatas que se colocam perante a implementação de um sistema de computação ubíqua.

Em vez da criação de sistemas que respondam por completo a todas as questões, i.e., ambientes físicos, interligados entre si, que implementem a visão de Weiser na sua totalidade, deve-se seguir por uma via prudente que seja iterativa, começando por identificar as plataformas (*hardware*, *software*, aplicações) com as quais se pode efectuar um trabalho consistente [16].

A computação ubíqua representa uma oportunidade de vulto para o acentuar das experiên-

---

<sup>2</sup>Consultar "*Embedded computing - the final definition*" em <http://www.embedded-computing.com/departments/foreword/2006/11/>

cias dos utilizadores de espaços físicos, sejam privados ou públicos (mais interessantes para a presente tese), através de meios efectivos que apoiem as suas actividades localizadas, podendo dar um suporte personalizado. Um cenário bem determinado para o desenvolvimento de sistemas de computação ubíqua, que sejam suportados por plataformas disseminadas, correntes, é o de acentuar as experiências dos utilizadores através do "aumento" dos objectos físicos, com os quais se cruzam e interagem no seu dia-a-dia, apresentando informação e serviços adicionais.

Para que as soluções adoptadas sejam realmente bem sucedidas nestes cenários, estas devem seguir uma aproximação que, discretamente, seja capaz de conjugar as actividades que decorrem no espaço físico com a nova funcionalidade de aumentação.

Existem duas tendências, algo conflituosas, no que toca a disponibilizar informação e serviços em ambientes colaborativos. Uma delas direcciona-se para a disponibilização através de ecrãs públicos situados para uma possível audiência vasta, enquanto que a outra explora a mobilidade de pequenos ecrãs em dispositivos móveis para oferecer conteúdos mais personalizados [17]. Torna-se claro que, enquanto ecrãs grandes (e partilhados) providenciam maiores escolhas, flexibilidade e apresentação de conteúdos, os dispositivos móveis oferecem uma maior diversidade em termos de personalização [18].

Os grandes ecrãs situados em ambientes públicos, pela sua natureza, são direccionados para a apresentação de conteúdos para atrair a atenção da audiência que passa junto a eles (e.g., notícias e informações de viagens mostradas em aeroportos e estações de comboios). Embora sejam úteis para dar informações a uma vasta audiência, não serão tão adequados para a oferta directa de serviços interactivos. Esta situação pode derivar do facto de ser difícil atrair a interacção do utilizador devido, em primeira instância, a factores tal como o embaraço social [19]. Enquanto que a provisão de informação personalizada pode ser um incentivo para o uso de ecrãs públicos interactivos, o esforço técnico adicional por parte do utilizador (e.g., com configurações e/ou autenticações) pode funcionar como dissuasor [20]. Para situações deste tipo, o envio de conteúdos para um dispositivo móvel pessoal poderá ser uma solução atractiva pois este pode já estar personalizado para o respectivo utilizador. Um sistema que conjugue as duas tendências poderá ser a solução em muitos cenários.

### 1.1.2 Objectivos

Com este trabalho, propõe-se alcançar um protótipo estável de sistema de computação ubíqua, baseado em tecnologias reconhecidas, com um objectivo applicacional bem focado.

Verifica-se que o paradigma da computação ubíqua é muito vasto e que a implementação de

sistemas que tornem os ambientes do dia-a-dia totalmente ubíquos não é uma tarefa simples e linear. Actualmente, são alguns os projectos conhecidos, desenvolvidos por parte de entidades de investigação de renome (e.g., Oxygen do MIT [21], HP Cooltown dos HP Labs [22], e Aura na CMU [23]), que arriscam uma implementação ambiciosa de sistemas vastos e complexos que apoiem a maior parte das tarefas quotidianas. Contudo, estes projectos também são compostos por sub-projectos mais específicos que implementam aplicações mais simples.

No caso presente, propõe-se uma infra-estrutura computacional ubíqua específica, que a um nível de abstracção elevado permita o fácil e rápido desenvolvimento de sistemas que aumentem objectos físicos. Estes objectos estarão presentes num ambiente, podendo o próprio espaço físico ser considerado, numa implementação mais restrita, como um grande objecto. O aumento dos objectos deverá ser feito através da apresentação de informação multimédia adicional em ecrãs públicos colocados, preferencialmente, junto aos respectivos objectos. Inicialmente, esta informação visará dar a conhecer algo mais sobre o objecto que vai além das formas visíveis, mas permitindo ainda o surgimento de novas funcionalidades agregadas à análise dos objectos em causa. A informação deverá ser contextual, baseando-se não só no objecto, mas também na sua localização, no espaço e no perfil dos utilizadores. Este último ponto terá um peso elevado na especificação da infra-estrutura.

Tem-se como objectivo elementar a distribuição do sistema pelas várias entidades presentes no ambiente, sejam utilizadores, dispositivos ou objectos. Pretende-se que o sistema seja o menos intrusivo possível e que aproveite ao máximo aquilo que um potencial utilizador já domine. Assim, a abordagem a seguir para a apresentação da informação deve ser [24]: feita onde e quando é útil, o mais discreta e menos intrusiva possível, e com interacção passiva. A infra-estrutura deverá ter ainda em conta uma segunda abordagem onde seja possível existir uma interacção activa, na qual o utilizador poderá fazer opções que determinem a informação a mostrar ou a activação de serviços.

O suporte applicacional da tese baseia-se num estudo, e consequentes resultados, retirado da implementação de dois sistemas que utilizam tecnologias diferentes. Em qualquer um dos sistemas, a apresentação da informação deve ser feita em ecrãs públicos, pretendendo-se utilizar a tecnologia RFID num dos casos e a Bluetooth no outro. Para o segundo, pretende-se também estudar e implementar um módulo da infra-estrutura no qual seja abordada a tendência de visualização de informação em dispositivos móveis pessoais, enquadrando-se na abordagem activa de interactividade (recentemente apresentado em [25]).

Pretende-se ainda que o levantamento do estado da arte quanto à Computação Ubíqua,

às tecnologias, e aos projectos relacionados seja o mais aprofundada possível, fazendo deste documento uma referência nas áreas que aborda.

## 1.2 Metodologia

Este trabalho foi realizado por fases, que se descrevem de seguida.

- **Fase 1 - Estudar e compreender o tema e respectiva área científica:** Estudo sobre a Computação Ubíqua de modo a levantar o conjunto de características que realmente a definem. Determinação dos projectos mais importantes nesta área. Enquadramento do tema na área quanto aos objectivos que visa alcançar.
- **Fase 2 - Pesquisar e analisar ferramentas e tecnologias de suporte ao trabalho:** Fase determinante a partir da qual se verificam quais são as tecnologias mais adequadas para a implementação de sistemas de Computação Ubíqua e, mais concretamente, para a especificidade do trabalho abordado pela tese.
- **Fase 3 - Pesquisar e analisar trabalhos relacionados:** Levantamento do estado da arte quanto a sistemas e projectos relacionados com o trabalho a desenvolver.
- **Fase 4 - Apresentar princípios e desenhar uma arquitectura de suporte:** Fase em que se projecta a infra-estrutura desejada com a definição dos princípios e requisitos elementares e com o desenho dos componentes da arquitectura.
- **Fase 5 - Implementar o primeiro protótipo de infra-estrutura:** Implementação de um sistema suportado pela arquitectura desenhada. Este primeiro protótipo implementa um cenário específico, utilizando a tecnologia de identificação RFID e ecrãs públicos.
- **Fase 6 - Reajustar a infra-estrutura genérica:** Com a implementação do protótipo foram feitas algumas alterações na arquitectura.
- **Fase 7 - Implementar o segundo protótipo com outra tecnologia base:** Segunda instanciação da infra-estrutura com a utilização da tecnologia de comunicação Bluetooth. Como teste, esta implementação visa um outro cenário, também com ecrãs públicos, mas com a adição de dispositivos móveis ao ambiente aplicacional.
- **Fase 8 - Reajustar o desenho da infra-estrutura:** Após a realização do segundo protótipo foram feitos mais alguns reajustamentos e adicionado um novo componente à arquitectura.
- **Fase 9 - Escrever a dissertação:** Elaboração do presente documento.

## 1.3 Mapa da Tese

Este documento apresenta o trabalho efectuado, integrando os sete capítulos seguintes:

- **Capítulo 1 - Introdução**

O presente capítulo apresenta, resumidamente, a dissertação, abordando as motivações e objectivos num enquadramento geral.

- **Capítulo 2 - Enquadramento Teórico**

No capítulo dois é apresentada a Computação Ubíqua, principal área científica onde pode ser inserido o trabalho desenvolvido, dando-se ênfase especial à noção de informação contextual.

- **Capítulo 3 - Trabalho Relacionado**

O capítulo três aborda tanto as principais tecnologias como os projectos de investigação e desenvolvimento que se relacionam com o trabalho da dissertação.

- **Capítulo 4 - Princípios e Arquitectura**

Este capítulo é dedicado à apresentação dos princípios em que se baseia o conceito apresentado. Também são descritos os requisitos e é apresentada a arquitectura com descrição de todos os módulos que a compõem.

- **Capítulo 5 - Realização do Protótipo divingForPearls**

A descrição do desenvolvimento de um protótipo implementado com RFID e linguagem C# é feita no capítulo cinco. Inicialmente, é apresentado o tipo de cenário ao qual a sua implementação é adequada.

- **Capítulo 6 - Realização do Protótipo directInfo2u**

Este capítulo tem uma estrutura idêntica ao anterior, sendo reservado para uma segunda implementação da infra-estrutura, neste caso com tecnologias Bluetooth e Java.

- **Capítulo 7 - Conclusões e Trabalho Futuro**

Por último, são apresentadas as conclusões, onde se inclui um estudo comparativo com os projectos apresentados no capítulo três, e o trabalho futuro.

## Capítulo 2

# Enquadramento Teórico

Este capítulo é dedicado ao aprofundamento dos conceitos nos quais se insere ou baseia o tema apresentado nesta dissertação, fazendo o enquadramento histórico e teórico que facilita a leitura dos capítulos seguintes. Assim sendo, verifica-se que a área da Computação Ubíqua e o conceito de Informação Contextual são os dois elementos base que definem os princípios e o desenvolvimento da infra-estrutura desejada.

### 2.1 Uma Visão

Derivada do latim *ubique*, a palavra ubíqua é um adjectivo que significa "que está em toda parte ao mesmo tempo"<sup>1</sup>. A conhecida expressão grega *hic et ubique* enfatiza o aqui e em toda parte. Mark Weiser<sup>2</sup> utilizou, ou melhor, "construiu" o conceito Computação Ubíqua<sup>3</sup> (Ubi-comp), ou Computação Pervasiva<sup>4</sup>, e apresentou-o em "The Computer for the 21st Century"[15] com os princípios "*making computing an integral, invisible part of the way people lives their lives*". Assim, as ideias propostas por Weiser abriram uma vasta área de investigação focada em aplicar a ideia de fazer com que os computadores se desvançam por entre aquilo que é o dia-a-dia das vidas, "percam as suas formas visíveis", "*weave themselves into the fabric of everyday life*", mas permanecendo interligados, em pleno funcionamento. Esta é uma grande mudança para a computação centrada no humano, onde a tecnologia deixa de ser uma barreira e adapta-se às

---

<sup>1</sup>Fonte: Dicionário Ilustrado da Língua Portuguesa da Editorial Verbo.

<sup>2</sup>Frequentemente considerado o pai da Computação Ubíqua, foi *chief technology officer* no centro Xerox's Palo Alto Research Center (Parc), tendo falecido em 1999. Mais informações em: <http://www2.parc.com/csl/members/weiser/>.

<sup>3</sup>Do anglo-saxónico *Ubiquitous Computing*.

<sup>4</sup>Do anglo-saxónico *Pervasive Computing*.

## 2.2. INTRODUÇÃO À COMPUTAÇÃO UBÍQUA

necessidades e preferências humanas, encontrando-se "nos bastidores" até que seja necessária. Este é um caminho em direcção a uma maior naturalidade na interacção e no uso do poder dos sistemas de computação de redes que assim não estarão ligados apenas à Internet ou a outros computadores, mas a pessoas, a locais, e a coisas e objectos do dia-a-dia. Em 1991, esta visão de uma computação fortemente embutida nas "coisas" parecia longe de ser alcançada, a sua viabilidade era questionável. No entanto, já em 1969 tinha surgido o romance "Ubik", de Philip K. Dick, que teve forte influência em Weiser. Este romance previa um futuro onde tudo, desde maçanetas de portas a autoclismos, estaria interligado de forma inteligente [26].

## 2.2 Introdução à Computação Ubíqua

Ainda hoje é difícil prever quais serão as aplicações da Ubicomp que se tornarão banais, tal como a rápida evolução das tecnologias Web também não era de fácil previsão nos primeiros tempos de desenvolvimento. É interessante olhar para a forma como Weiser olhou para as diferentes fases da computação e previu uma possível evolução da Ubicomp. As três ondas da computação (ver Figura 2.1, imagem da esquerda) separam a era da *Mainframe* (um computador usado por muitas pessoas) da era do *Personal Computer* (PC) (um computador para uma pessoa) e da era da Ubicomp (uma pessoa partilha vários computadores).

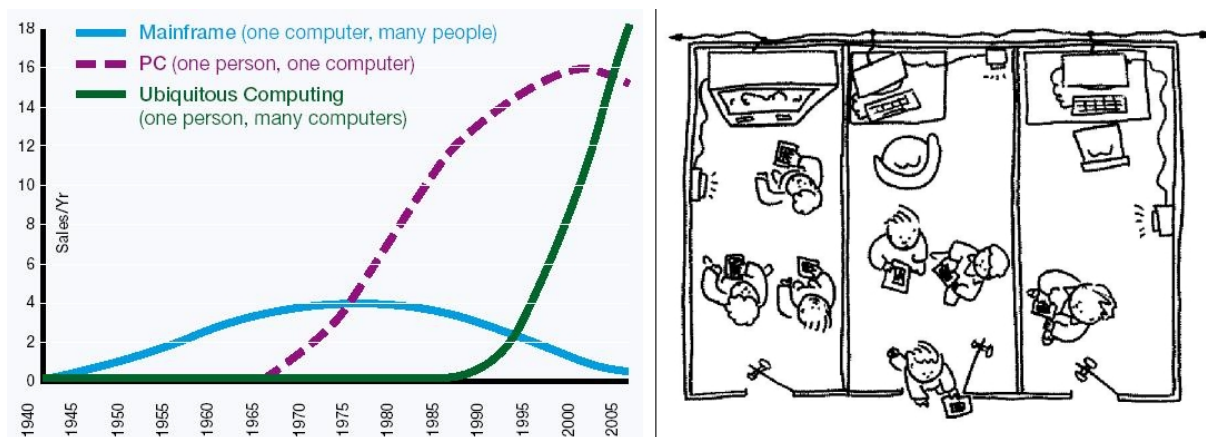


Figura 2.1: (esquerda) As três ondas da computação [1]; (direita) ParcTab: Um sistema de computação *context-aware* [2].

Foi sensivelmente há 15 anos que o produtivo artigo de Weiser instigou a imaginação colectiva a procurar o desenvolvimento de sistemas no campo Ubicomp. Contudo, apenas recentemente têm aparecido investigadores a desafiar os preceitos dessa visão e a explorar novos domínios de aplicação, iniciando um debate na área [27, 28]. Quanto às previsões de Weiser, estas não

estiveram longe da realidade pois tem sido crescente o aparecimento de trabalhos, sobretudo de investigação, para que o dia-a-dia humano possa ser mais assistido, suportado e melhorado. Têm surgido trabalhos de muita qualidade em áreas intimamente ligadas à Ubicomp, tais como sistemas de informação contextual (*context-aware*), redes de sensores, *toolkits* para a construção flexível de aplicações, espaços inteligentes (*Smart Spaces*<sup>5</sup>), entre outros. Muito daquilo que ainda hoje se discute já era apresentado em 1994 por Schilit et al. quando descreviam sistemas que examinavam e reagiam às alterações de contexto de um indivíduo [2] (ver Figura 2.1, imagem da direita).

Tal como sugeriu Weiser em 1991, uma implementação Ubicomp deve integrar três tipos de entidades. Estes são: 1) os agentes de utilizadores (i.e., dispositivos carregados pelos utilizadores), 2) os dispositivos integrados no meio ambiente, 3) e a Internet, com a interacção digital entre estas entidades a dar origem a novas aplicações. Com a crescente maturação das tecnologias, em que existe um maior desempenho para um menor tamanho de dispositivos, tais como processadores, memórias, tecnologias de rede, ecrãs e sensores, caminha-se mais facilmente em direcção à Ubicomp, a uma conectividade ubíqua e a interfaces mais adaptáveis.

Já são muitos os objectos e os dispositivos que possuem processadores e sensores embutidos. Os carros são talvez o melhor exemplo, possuindo sensores que possibilitam as mais variadas respostas em diversas situações, sobretudo ao nível da segurança. De todos os modos, estes sistemas são, regra geral, considerados *stand-alone* e não interagem necessariamente com outros objectos. Uma forma primária de redes de comunicação e informação ubíquas aparece na vasta expansão do uso de telemóveis com o número total de dispositivos à escala mundial a ultrapassar os três mil milhões em Julho de 2007 [29]. Estes dispositivos, juntamente com os PDAs (*Personal Device Assistants*), têm vindo a ocupar um papel integrante no quotidiano das pessoas, inclusive até mais que a Internet.

Hoje em dia, já são vários os projectos em desenvolvimento para que rapidamente se chegue ao próximo estágio. Através da colocação de transceptores (transmissores/receptores) móveis de curto alcance em dispositivos adicionais e em outros objectos do dia-a-dia podem-se estabelecer outras formas de comunicação. Pretende-se que as ligações ou comunicações não sejam estabelecidas apenas entre as pessoas ou entre pessoas e computadores, mas que se avance para o tipo pessoas-objectos, e mais especificamente para objectos-objectos. Foi adicionada uma nova dimensão ao mundo das Tecnologias de Informação e Comunicação (ICTs), onde no conceito "*anytime, anyplace*" com conectividade para qualquer um, "*anyone*", passou-se para conectivi-

---

<sup>5</sup>A Computação Ubíqua é também conhecida por esta expressão pela inteligência que confere aos locais.

## 2.2. INTRODUÇÃO À COMPUTAÇÃO UBÍQUA

dade para qualquer coisa, "anything" [3] (ver Figura 2.2).

Uma nova rede de redes, conhecida por *Internet of Things*<sup>6</sup>, surge com a multiplicação das ligações disponíveis entre todas as entidades consideradas num espaço UbiComp. Este conceito de "Internet das coisas" não pode ser considerado ficção científica nem desajustado daquilo que se pode praticar actualmente na indústria dado ser baseado nos avanços sólidos das tecnologias e nas visões das redes ubíquas [3].

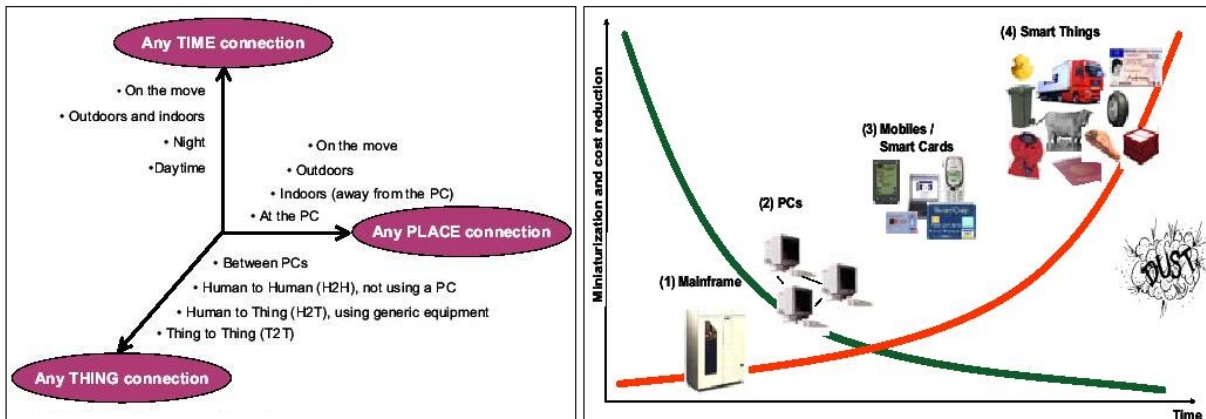


Figura 2.2: (esquerda) A nova dimensão no mundo das ICTs; (direita) Miniaturização em direcção à *Internet of Things* [3].

Este conceito de *Internet of Things* representa, para muitos, o futuro da computação e das comunicações, ou interações. A sua evolução está dependente de uma inovação constante em campos como as redes de sensores sem fios e a nanotecnologia. Esta última possibilita que entidades cada vez mais pequenas, e com menores custos, possam conectar-se e interagir entre si [3] (ver Figura 2.2).

Quanto às interações, estas podem ser de dois tipos: explícitas ou implícitas. As primeiras são as mais vulgares, onde uma acção consciente por parte de um utilizador possibilita a interacção. As implícitas são automáticas e podem acontecer sem a intervenção directa, consciente, por parte do utilizador. São estas últimas as mais desejadas, as que mais contribuem para a mudança real em direcção à UbiComp.

A UbiComp é transversal a quase todas as áreas das tecnologias de informação, estando o sucesso da sua aplicação dependente de factores essenciais. Segundo Ley [30] devem considerar-se factores nas seguintes áreas:

- Interoperabilidade - produção de normas para redes e dispositivos, detecção automática

<sup>6</sup>*Internet of Things* é um outro termo pelo qual é muitas vezes conhecida a Computação Ubíqua.

de rede e dispositivo, configurações próprias, etc;

- Miniaturização - diminuição do que envolve as tecnologias *wireless*, de sensores e de processadores;
- Conectividade *wireless* - desenvolvimento e aperfeiçoamento das tecnologias na área;
- Interfaces inteligentes - agentes inteligentes, interfaces naturais, melhoria nas tecnologias para visualização, etc;
- Sistemas inteligentes - redes de sensores, noção do contexto, localização, redes semânticas;
- Segurança e fiabilidade - sistemas seguros e de confiança, questões de privacidade.

O conceito inicial de UbiComp, em [15] e [31], não determina instruções precisas para a implementação de sistemas concretos, ou seja, não apresenta um modelo a partir do qual se possa efectuar uma construção directa dos sistemas. A UbiComp propõe um paradigma de desenvolvimento, uma "filosofia". Apesar da sua difusão por várias disciplinas (e.g., as redes de sensores, a computação em GRID, a móvel e a embutida) torna-se possível apresentar um conjunto de características que têm vindo a surgir ao longo do processo de maturação da UbiComp. A caracterização da UbiComp passa obrigatoriamente por características que podem ser encontradas em sistemas e aplicações da computação, tais como a descentralização, a mobilidade, a interconectividade e a portabilidade.

### 2.2.1 Questões Pertinentes em Computação Ubíqua

Com a evolução da UbiComp será cada vez mais difícil lidar com a vasta quantidade de informação produzida por sistemas deste tipo. As grandes quantidades de dados produzidas acerca dos utilizadores, dos objectos e dos cenários vão requerer novas formas de tratar, pesquisar e apresentar a informação. Primeiramente, serão necessárias novas aplicações para tirar partido dos dados recolhidos em tempo real. A computação em GRID, tendo surgido com vista ao tratamento de aplicações intensivas de dados, pode ser uma boa solução para os sistemas de UbiComp [32, 33, 34].

Também são necessárias novas arquitecturas e estruturas de dados, escaláveis e adaptáveis, que façam frente ao requisitos de armazenamento e de enorme processamento. Espera-se que os sistemas fiquem, progressivamente, em condições de responder positivamente à entrada de dados a partir do mundo real para que possam executar as acções apropriadas, cada vez mais, sem a intervenção humana. Os sistemas poderão apresentar a informação depois de uma correcta filtragem e análise dos dados e de acordo com o ambiente onde se integram.

A capacidade que máquinas e sistemas tenham para interrogar outras máquinas e sistemas e

para partilhar informação será uma chave para a implementação da visão da computação ubíqua. Uma solução muito focada passa pela *semantic web*<sup>7</sup>, com referências em *Task Computing* [36, 37, 38].

A UbiComp torna-se mais invasiva e persistente que, por exemplo, a Internet, uma vez que tem por base de funcionamento uma interacção implícita, podendo gerar uma grande quantidade de informação sobre a localização e actividade do utilizador. Assim sendo, a privacidade é uma questão ainda mais preponderante neste tipo de computação onde muita da informação recolhida é sobre os utilizadores. Num sistema computacional que se pretende "invisível" e extensível torna-se difícil saber o que controla o quê, o que está ligado a quê, para onde se desloca e como está a ser usada a informação, e quais são as consequências após uma qualquer acção ter sido executada [39].

Existem também questões respeitantes ao impacto social e à conveniência por parte das tecnologias pervasivas. Potencialmente, estas tecnologias poderão melhorar a segurança em sociedade, diminuir o fosso digital entre pessoas, e até contribuir para a longevidade populacional. No entanto, os benefícios devem ser pesados contra os custos financeiros e de privacidade que acarreta um sistema de UbiComp.

## 2.3 Elementos-Chave num Ambiente de Computação Ubíqua

A implementação de um ambiente de UbiComp deve passar pela análise dos seguintes pontos elementares [30]:

- **Localização**

Os sistemas avançam para um outro nível de inteligência quando objectos e dispositivos computacionais têm informação sobre a respectiva localização. Deste modo, permitem que pessoas, recursos e objectos sejam descobertos e ainda definem serviços e ferramentas adicionais para localização. Os serviços de localização encontram-se em grande crescimento tendo uma importância enorme nos tempos que correm. Os sistemas de posicionamento global conhecidos por GPS (*Global Positioning System*) são disso um grande exemplo, esperando-se que em 2010, segundo a IMS Research<sup>8</sup>, sejam 70 milhões os telemóveis com GPS integrado existentes na Europa [40].

---

<sup>7</sup>Serviços *Semantic Web* são, geralmente, definidos como a aumentação de descrições de *Web Services* através de anotações semânticas, dando um maior grau de automação à descoberta, composição, invocação e monitorização de serviços num ambiente aberto, não regulamentado e, muitas vezes, caótico [35].

<sup>8</sup><http://www.imsresearch.com/>

São várias as tecnologias (ver Subsecção 3.3) que podem adicionar aos objectos e aos dispositivos, e conseqüentemente às pessoas, a noção de localização. Esta pode ser estabelecida de várias formas e com vários níveis de exactidão. A tecnologia RFID (*Radio Frequency Identification*) pode ser utilizada em cenários mais contidos e fechados nos quais se marcam os locais onde se procura fazer detecção e localização de entidades (e.g., objectos e pessoas). A tecnologia Bluetooth também tem aplicações semelhantes, existindo casos de utilização em espaços exteriores [41]. Já os sistemas RTLS (*Real Time Locating Systems*) permitem a localização de entidades com um razoável grau de precisão e são mais úteis em cenários onde interessa conhecer a localização da entidade a qualquer momento e em qualquer ponto. Quanto à localização mais precisa em espaços exteriores não contidos, o grande desenvolvimento de sistemas baseados em tecnologias de posicionamento ou navegação por satélite tem possibilitado uma melhor cobertura dos espaços.

- **Identificação**

A identificação, tanto de objectos como de dispositivos, é importante para que numa rede ubíqua, num ambiente inteligente, estes possam ter um papel útil através da sua identidade única. Este é um elemento-chave porque permite que os objectos vulgares do dia-a-dia sejam recursos num ambiente computacional e que possam servir como interfaces para outros recursos. A problemática da correcta identificação torna-se num ponto fulcral da investigação na área da Ubicomp. Muitas das tecnologias usadas para localização também servem para a identificação. Este elemento pode usar, por exemplo, RFID, *tags* visuais (*visual barcodes*) ou Bluetooth (ver Subsecção 3.3).

- **Monitorização Sensorial**

A adição deste elemento aos anteriores pode dotar os sistemas de recursos adicionais, criando redes inteligentes que podem recolher os mais variados dados e responder a diversos estímulos. Com a ajuda de sensores, um ambiente de Ubicomp detecta e responde a eventos, a situações, sem que para tal tenha que existir uma interacção humana propositada. Os sensores podem medir a temperatura, a pressão, a qualidade da água, entre outras. Com sensores tipo *wireless* as redes de sensores podem ser rapidamente instaladas, podendo utilizar *web services* para se integrarem com outros sistemas de informação. Sensores de inércia têm sido embutidos em telemóveis e controladores de jogos, como no caso da Nintendo Wii, para permitirem que o utilizador possa interagir com o dispositivo através do

movimento. Por outro lado, os MEMS<sup>9</sup> (*Micro-Electro-Mechanical Systems*) são usados, por exemplo, em carros para detectarem colisões e responderem com o disparo de *airbags*.

- **Conectividade**

Os meios de comunicação são essenciais para o estabelecimento de interações e cooperações entre entidades distribuídas num espaço Ubicomp. Este é um elemento que deve dar primazia às tecnologias *wireless*, pois estas são indicadas para os dispositivos portáteis/móveis. Encontra-se intimamente ligado aos pontos anteriores e tecnologias abordadas na identificação e localização são também usadas para conectividade. Na subsecção 3.2, são apresentadas tecnologias preponderantes nos dias de hoje.

- **Interação Pessoa-Máquina (HCI)**

É desejável que existam mudanças profundas relativamente às formas tradicionais de interação entre humanos e computadores (HCI<sup>10</sup>). Por um lado, as tecnologias ligadas à Ubicomp já providenciam as fundações técnicas para dispositivos de entrada e saída avançados e para sistemas embutidos de pequena escala. Por outro lado, os objectos físicos devem ter um papel central ao nível das representações físicas e dos controlos para a informação digital. Isto é reflectido numa nova vaga de investigação nas HCI em que se procuram formas para ligar os mundos físico e digital [42]. Tem como objectivos a exploração da relação entre a representação física e a informação digital, e trazer à discussão novas formas de interação que não são encontradas nas infra-estruturas existentes. Estas interfaces pessoa-computador são usualmente referidas como sendo do tipo *graspable* [43] ou interfaces tangíveis [44].

Com os desenvolvimentos na área da Ubicomp, torna-se possível integrar computação e comunicação em praticamente todos os tipos de objectos e dispositivos. Deste modo, é possível encontrar-se num cenário quotidiano cada vez mais objectos e dispositivos inteligentes que podem interagir entre si ou com os utilizadores. É natural que as interfaces para interações com objectos sejam, geralmente, bem diferentes das encontradas nas relações com os computadores convencionais [45]. Segundo [46], o crescente desejo pelas interfaces mais naturais torna-se notório pelo surgimento de muitas sugestões para interfaces manipulativas e hápticas (e.g., o projecto *Mediacup* [47]), e pelos paradigmas de interação que

---

<sup>9</sup><http://www.memsnet.org/mems/>

<sup>10</sup>HCI - Do anglo-saxónico *Human Computer Interaction*, é o campo de investigação que trata as questões relacionadas com as interfaces pessoa-máquina e os paradigmas de interação.

têm vindo a ser apresentados ao longo da última década, tais como em [48], [49] e [50].

- **Noção do Contexto**

Num sistema ubíquo, um dos principais objectivos, senão o mais importante, é o de facultar informação relevante sempre que necessária num determinado momento e num dado local. Esta informação, que deve ser apresentada na forma mais adequada para a ocasião, poderá ser considerada *"just in time"* se objectos e dispositivos conhecerem a sua localização e o ambiente em que se encontram, podendo detectar e reconhecer as pessoas, além de automaticamente descobrirem outros dispositivos, recursos e serviços. Os sistemas podem assim adaptar as suas saídas, aquilo que oferecem, a um conjunto de características únicas. Esta noção de contexto é fundamental para a adaptação e personalização dos sistemas de informação ubíquos que se pretendam manter discretos até que sejam necessários, dando-se aí pela sua presença.

Os sistemas *"context-aware"*<sup>11</sup> auxiliam o processo de filtragem de informação em vez de terem que ser os utilizadores a procurar aquilo que necessitavam para um dado momento, e de acordo com o cenário. Este componente de consciência do contexto é fundamental para o conceito de Ubicomp ao permitir que os utilizadores de um sistema não necessitem de se concentrar na tecnologia utilizada para obtenção da informação, podendo focar completamente a sua atenção na tarefa a realizar. Enquanto esta é realizada, a informação que for relevante e adequada para o contexto deverá ser automaticamente disponibilizada e de uma forma que seja o menos intrusiva possível.

## 2.4 Informação Contextual

O conceito de informação contextual (ou *context-aware*) derivou da ideia inicial da Ubicomp, onde um sistema tem a noção do contexto onde está inserido e age em conformidade com o mesmo. Esta interpretação de contexto, por parte das aplicações computacionais, ganhou interesse redobrado com o desenvolvimento dos dispositivos móveis (e.g., telemóveis, PDAs e *smart phones*). Em [51], Chen e Kotz, conduzem uma pesquisa sobre investigação em projectos e aplicações que usam a noção de contexto, dando realce à forma como recentes avanços nos dispositivos móveis, conjuntamente com as redes sem fios, criam um cenário propício para uma

---

<sup>11</sup> *Context-Aware* é um outro termo pelo qual a Computação Ubíqua é reconhecida, sendo um conceito elementar intimamente ligado à mesma.

renovada expressão no que toca a sistemas de informação contextual. Esta evolução dos dispositivos traz desenvolvimentos promissores com os serviços de informação contextual a poderem ter em conta a localização do utilizador, as suas preferências, e os objectos com os quais se cruza.

Diferentes definições de contexto geram grandes diferenças de aproximação, sendo o conceito muitas vezes descrito de acordo com informação sensorial ou como sendo mais abrangente ao corresponder a todo um ambiente. Embora seja um elemento preponderante em Ubicomp ainda não existe uma definição padrão para contexto nesta área.

Schilit e Theimer [52] foram, eventualmente, os primeiros a usar o conceito de contexto e a sua noção, definindo-a como a capacidade patenteada por aplicações para encontrar e reagir a alterações no ambiente em que se situam. Sendo uma definição muito genérica, ainda a estreitaram com a definição de informação contextual como sendo o conhecimento da localização, das identidades de pessoas e objectos próximos, e das mudanças nesses objectos. A localização geográfica é de relevância estratégica. Em [51] é apresentada uma lista de sistemas que auxiliam na obtenção de informação contextual e quase todos eles recorrem a algum tipo de mecanismos de localização. A localização ajuda claramente na interpretação contextual sendo possível enquadrar a entidade (utilizador, dispositivo, objecto) em relação a outras entidades.

Mais tarde, Chen [51] complementou a proposta de Schilit em [2], aparecendo a informação contextual dividida de acordo com sub-áreas [53]:

- Contexto computacional - Recursos, redes, conexões, capacidades, etc.
- Contexto humano - Perfil, localização, estado emocional, etc.
- Contexto físico - Temperatura, iluminação, objectos, etc.
- Contexto temporal - Dia, mês, estação do ano, históricos, etc.

Há quem se limite a definir informação contextual de forma muito genérica como sendo qualquer informação capaz de caracterizar a situação de uma entidade, mas já Schmidt et al. em [54] restringem essa noção baseando-se em [52]. Barkhuus et al. apoiam-se também na definição original de Schilit e Theimer, mas, contudo, referem que como o ambiente e a situação são frequentemente usados como sinónimos para contexto, os conceitos apenas definem a noção geral de informação contextual. Assim sendo, o contexto é usualmente definido por uma categorização de contexto e de informação sensorial, que é a essência da noção de contexto [4]. Não existindo uma diferenciação clara entre informação contextual e informação sensorial, o modelo de informação contextual (ver Figura 2.3) foi proposto com a finalidade de facilitar o desenho de aplicações a partir de uma quantidade limitada de medidas de contexto [4]. A consideração de medidas

individuais pode ajudar na criação de aplicações interactivas usáveis que podem suportar, por exemplo, trabalho colaborativo. O modelo apresentado por Barkhuus suporta aplicações reais que considerem um número limitado de medidas sensoriais.

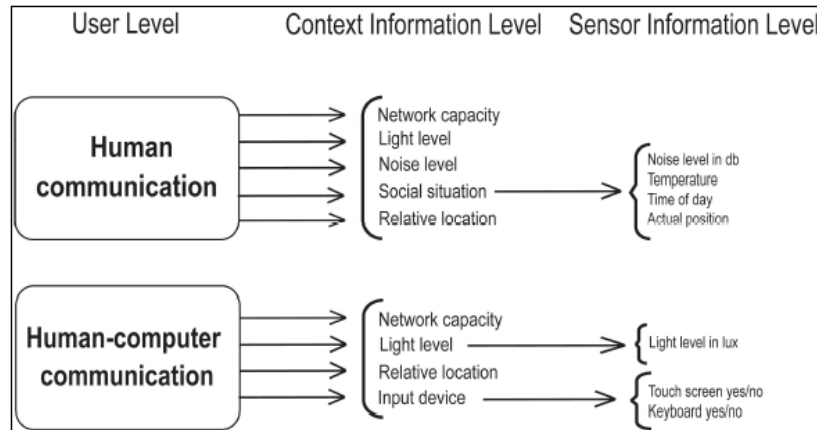


Figura 2.3: Um modelo para Informação Contextual [4].

Os agentes<sup>12</sup> são um elemento considerado importante numa conjugação com a informação contextual. Podem fazer com que os sistemas se adaptem e respondam de forma inteligente aproximando-se do conceito da Ubicomp. Integrando agentes, os sistemas podem responder a dados ambientais, mas também o podem fazer relativamente a dados sobre o utilizador e a suas preferências, ou o seu perfil, em conjugação com a localização, a hora, o dispositivo ou objecto a ser usado, ou até mesmo o que o utilizador está a fazer.

Uma outra questão fundamental para a Ubicomp e que se relaciona com a noção do contexto é a capacidade crescente de dispositivos e sistemas para descobrirem ferramentas e serviços de modo automático. De uma forma mais simples e directa, esta procura pode levar, por exemplo, ao uso de impressoras ou de ecrãs, mas espera-se que o uso mais relevante se coloque na descoberta e seguimento do utilizador (ou dispositivo que o identifique de forma unívoca) por parte da informação. O seguimento pode ser feito através do movimento dos conteúdos de dispositivo (sobretudo móvel) para dispositivo ou de ecrã para ecrã, nas proximidades do utilizador, e enquanto este se move (e.g., da casa para o carro, entre divisões num edifício, ou até mesmo numa avenida). Actualmente, já é possível encontrar sistemas que de alguma forma implementam o que foi referido, como os exemplos de ligações automáticas a *hotspots* Wi-Fi e de serviços telefónicos do tipo *follow me*<sup>13</sup>.

<sup>12</sup>Para saber mais sobre agentes: <http://agents.umbc.edu/>

<sup>13</sup>"Siga-me" é um tipo de serviço disponibilizado pelas empresas prestadoras de serviços telefónicos que permite a transferência de chamadas de um número telefónico para outro, onde o cliente poderá ser encontrado

A Computação Afetiva é uma outra área a considerar no desenho de sistemas de Ubicomp. Aqui, o contexto também estará relacionado com a detecção da atenção de uma pessoa e do seu estado emocional, podendo traduzir-se na adaptação dos sistemas aos estados emocionais dos seus utilizadores. A análise de voz, já usada, por exemplo, em *call centers*, o rastreo do olhar<sup>14</sup>, a análise de expressões faciais<sup>15</sup> [55], a localização e a forma como um utilizador interage com o sistema podem dar pistas acerca do seu estado emocional, permitindo uma análise à sua receptividade.

## 2.5 Projectos e Sistemas de Computação Ubíqua

Nesta secção são apresentados os projectos e sistemas de Ubicomp considerados mais relevantes para a evolução da área. Alguns pertencem ao passado e eram limitados, mas foram importantes para o início do sonho de Weiser. A maior parte do trabalho inicial na área da Ubicomp foi realizado com o propósito de demonstrar a sua viabilidade e possibilidades através do desenvolvimento de pequenos dispositivos computacionais, de redes *wireless*, ou através da programação de *software* para os dispositivos.

Com a implantação de tecnologias *wireless*, tais como Wi-Fi e Bluetooth, iniciou-se uma nova era nos desenvolvimentos de projectos de Ubicomp. Muitos dos projectos mais recentes baseiam-se nestas tecnologias e concentram-se muito em aspectos práticos como a usabilidade, as interfaces pessoa-máquina (HCI), as infra-estruturas *middleware* para suporte, ou os estudos de viabilidade de novos domínios de aplicação.

Salienta-se ainda o facto de muitos considerarem sistematicamente a ausência de uma infra-estrutura comum geral, que sirva de base de sustentação, como sendo um dos obstáculos chave à implementação de sistemas de Ubicomp sólidos e viáveis [56, 57, 58]. As particularidades de cada espaço e cenário de aplicação obrigam às mais diversas aproximações.

Um dos esforços iniciais relevante para a investigação de sistemas de Ubicomp foi o sistema de localização **Active Badge** [59]. Este foi originalmente concebido, desenhado e prototipado entre 1989 e 1992 nos laboratórios AT&T. O Active Badge (ver Figura 2.4) é (existe uma versão actualizada) um pequeno dispositivo que pode ser usado num edifício.

Cada um dos Active Badge possui informação com uma identidade única, sendo esta trans-

---

temporariamente.

<sup>14</sup><http://www.almaden.ibm.com/cs/projects/eyegaze/>

<sup>15</sup><http://affect.media.mit.edu/projectpages/esp/>

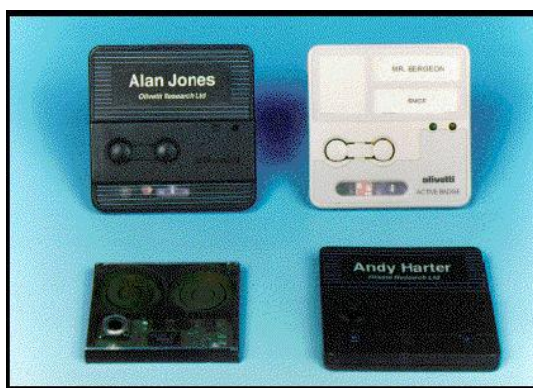


Figura 2.4: 4 gerações do Active Badge [5]: (esquerda em baixo) Primeira versão (código de 5 bits), (direita em baixo) segunda versão (10 bits), (esquerda em cima) terceira versão e (direita em cima) quarta e actual versão (48 bits, bi-direccional e com microprocessador 87C751).

missível via tecnologia InfraRed (IrDA<sup>16</sup>) a cada quinze segundos e com uma duração de um décimo de segundo. A vantagem da adopção da tecnologia IrDA estava na produção de componentes muito pequenos e baratos.

São colocados sensores nas várias divisões de um edifício de forma a detectarem os sinais emitidos pelos *badges*. Visto os sinais IrDA serem reflectidos pelas paredes então o sistema deve estar confinado a divisões para localizar os *badges*. Os sensores não necessitam de estar perfeitamente alinhados com os emissores dos sinais porque estes ressaltam das paredes e objectos. Com a colocação de sensores em posições altas em paredes ou nos tectos, e em todas as divisões, corredores e entradas e saídas, será possível determinar a localização de um *badge* em qualquer momento.

Alguns dos cenários iniciais pensados para o uso do sistema seriam a possibilidade de as portas abrirem automaticamente ao ser detectado um utilizador com credenciais para tal, as salas ou divisões poderiam cumprimentar as pessoas pelo seu nome, as chamadas de telefone poderiam ser automaticamente passadas para as divisões onde se encontrava o destinatário da chamada, e os terminais de computadores poderiam aceder de forma automática às preferências do utilizador registado no sistema [59, 15, 60].

Entre 1988 e 1992, a Xerox's PARC trabalhou num projecto que tinha como objectivo desenvolver dispositivos de teste que sustentassem uma infra-estrutura de Ubicomp. Desenvolveram então três tipos de protótipos com diferentes escalas: *tabs*, *pads* e *boards*. Estes são nomes genéricos, tendo o grupo produzido um protótipo para cada um dos tipos, aos quais chamou,

<sup>16</sup>Do anglo-saxónico *Infrared Data Association*.

respectivamente, **ParcTab**, **Mpad**, e **Liveboard**. O *board* foi o primeiro protótipo a ficar concluído, e todos eles ficaram funcionais e a serem usados em experiências no dia-a-dia do grupo [61].

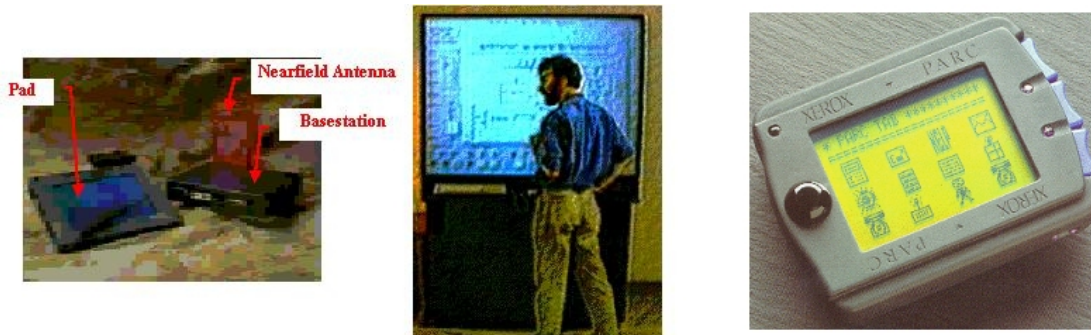


Figura 2.5: (esquerda) ParcPad, (centro) Liveboard e (direita) ParcTab da Xerox's PARC [1].

Os *tabs* eram suficientemente pequenos para serem presos a coisas, podendo ser de várias formas, e funcionando tal como uma nota *post-it* activa. Os *pads* eram vistos como folhas de papel para rascunho e não tinham uma identidade estática, acomodando várias tarefas. Poderiam ser usados para organizar tarefas numa secretária, avisando sempre que uma tarefa fosse concluída. Não foram feitos com a intenção de serem transportados. Os *pads* estariam nas divisões e um utilizador deixava o que estivesse a usar numa divisão antes de sair, evitando-se assim que os *pads* fossem carregados entre divisões. Com o Liveboard pretendia-se cobrir grandes áreas como as paredes e operações interactivas de "conversação" *wireless*, em que cada utilizador poderia escrever sem tocar no *board* [39]. O *board* serviria para a mediação de encontros ou reuniões, e até para reuniões entre o grupo PARC (nos E.U.A.) e o EuroPARC (na Europa). Um *board* no qual informação pública é visualizada pode adaptar-se aos seus leitores direccionando a informação para os mesmos. Isto era alcançado via Active Badges que os utilizadores carregavam e através de sensores colocados junto ao *board*.

Após a realização dos primeiros protótipos, são muitos os projectos que ao longo dos anos têm surgido com os mais variados objectivos. Alguns são apenas projectos de investigação, mas outros deram o salto tornando-se sistemas em utilização. Alguns são infra-estruturas complexas que visam suportar um ambiente Ubicomp fazendo a gestão das interacções entre entidades e serviços disponíveis, e outros são aplicações directas de algumas características elementares da Ubicomp. Alguns dos casos mais interessantes para a consolidação da Ubicomp, e até relevantes pelo número de vezes referenciados noutros trabalhos, são os que se descrevem de seguida.

O **Active Maps** [52], de 1994, foi o primeiro trabalho a utilizar explicitamente o conceito de

informação contextual. Como contexto foram utilizadas informações como localização, identificação de entidades (pessoas e objectos) próximas, bem como mudanças que ocorrem sobre estas entidades ao longo do tempo. O projecto foi desenvolvido por Bill Schilit, no centro Xerox's PARC, tendo sido utilizados, por exemplo, Active Badges ligados aos objectos para determinar a localização destes. O sistema Active Map pode ser visto como um sistema de publicação/-subscrição, baseado na localização, para a disseminação de informação contextual. Com este sistema, informação contextual é publicada num servidor Active Map, que depois a disponibiliza para as aplicações clientes.

Um projecto que ambiciona englobar todas as características fundamentais da UbiComp surgiu pelos laboratórios do MIT com a designação de **Oxygen** [21]. O projecto visa o suporte das actividades humanas através da configuração automática e invisível de componentes de *software* e *hardware* embutidos numa infra-estrutura de UbiComp. Sendo este um projecto muito ambicioso, conta com fundos por parte da DARPA (Defense Advanced Research Projects Agency) norte-americana e com um consórcio formado por várias companhias privadas, entre elas a Acer Group, a Philips, a Delta Electronics e a Hewlett-Packard. A arquitectura do Oxygen integra um conjunto de oito tecnologias diferentes com as seguintes designações: *Handy21*, *Enviro21*, *Network21*, *Conversational I/O*, *Web bots*, *Task automation*, *Recorder*, e *Specialized microprocessor*.

O projecto **Gaia** [62] leva as funcionalidades de um sistema operativo para espaços físicos. O Gaia alarga os conceitos típicos em sistemas operativos para incluir noção de contexto, localização, dispositivos de computação móveis e actuadores. O projecto investiga como construir aplicações de forma genérica, não fazendo qualquer assunção sobre a parametrização do *hardware* corrente num espaço. As aplicações devem poder ser construídas e depois colocadas em espaços com diferentes configurações, utilizando os recursos que se encontrem disponíveis. Este projecto releva a interacção entre espaços "activos" e utilizadores, com estes a interagirem com múltiplos dispositivos simultaneamente e a poderem programar os comportamentos das aplicações com base nos atributos do contexto.

São vários os projectos de UbiComp desenvolvidos pelo grupo de sistemas de Mahadev Satyanarayanan<sup>17</sup> na Carnegie Mellon University, sendo o **Aura** [23] o mais interessante. Este visa implementar um sistema que providencia uma "auréola invisível" (ou "aura") de serviços computacionais e de informação que persistam independentemente de onde se localize o utilizador, que se pode mover entre diferentes ambientes. A definição de ambiente é similar à de espaço

---

<sup>17</sup><http://www.cs.cmu.edu/~satya/>

activo no Gaia, correspondendo ambas as noções a espaços de Ubicomp. O Aura define uma infra-estrutura de *software* para suporte das tarefas dos utilizadores entre ambientes, maximizando os recursos existentes e tentando minimizar a distração por parte dos utilizadores. Esta é a diferença de fundo para o projecto Gaia, já que para este a noção de programabilidade do espaço por parte do utilizador era a ideia base do sistema.

Um outro projecto relevante é o **eClass** [63], que tem como missão o estudo da captura automatizada de experiências ao vivo para posterior acesso e consulta. O projecto iniciou-se em 1995 sob coordenação do professor Gregory Abowd no grupo FCE (Future Computing Environments) da Universidade Georgia Tech, onde se desenvolveram projectos como o Smart Floor [64] e o the Aware House [65]. O estudo do eClass foi direccionado para o contexto educacional, mas não de forma exclusiva. O objectivo do eClass é auxiliar tanto docentes quanto alunos, na sua actividade regular, através do uso de tecnologias reconhecidas. Com o sistema, os alunos podem tomar mais atenção às explicações por parte dos docentes, pois não necessitam de estar constantemente a apontar as matérias, sendo estas tiradas por um conjunto de programas que compõem o sistema. A infra-estrutura física do sistema é composta por um *whiteboard* electrónico, por projectores e por equipamento áudio e vídeo para captura da experiência em tempo real. Por outro lado, Zen\* é o nome dado ao sistema composto pelos servidores e aplicações responsáveis pela introdução dos conteúdos, captura da aula, disponibilização da aula, entre outras funcionalidades.

O DATAFOUNTAIN [66] é um dos primeiros projectos a implementar com sucesso o conceito de "*calm technology*", definido por Weiser em [67]). Este sistema é constituído por uma fonte, com três repuxos de água, ligada em tempo real a taxas de câmbio (entre o Yen, o Euro e o Dólar) presentes na Internet. A informação é actualizada de cinco em cinco segundos. Neste projecto, os habituais ecrãs são substituídos por algo mais calmo e natural para o ambiente em que se insere. Com o DATAFOUNTAIN pretendeu-se que a tecnologia de informação não fosse intrusiva, sendo colocada no ambiente de forma implícita, mas sem estar no centro da atenção. A fonte é equipada com um controlo que consegue fazer variar a altura dos jactos de água. As bombas são controladas através de uma ligação Ethernet e de um modulador de frequência.

## Capítulo 3

# Trabalho Relacionado

Após o aprofundamento dos conceitos fundamentais, neste capítulo apresenta-se o trabalho relacionado com a mesma. Dá-se a conhecer o estado da arte quanto a tecnologias que são vulgarmente associadas à Ubicomp, e que podem ser indicadas para a implementação de infra-estruturas como a apresentada neste trabalho. Apresenta-se também uma selecção de sistemas e projectos que contêm pontos em comum com o tema específico desta tese.

### 3.1 A Computação Embutida e as Tecnologias

Os contínuos avanços em termos da microelectrónica e da Computação Embutida levam a um grau de miniaturização da tecnologia computacional que permite o desenvolvimento e a produção de microprocessadores, memórias, tecnologias de comunicação *wireless*, sensores e baterias com forma e peso cada vez menores. Em conjunto com novos dispositivos de entrada/saída, estas tecnologias podem ser consideradas como sendo aquelas que possibilitam o avanço na realização de aplicações e sistemas de Ubicomp.

Devido a um menor tamanho, a um menor consumo de energia, e a preços menores, os microprocessadores, os módulos de memória e de comunicação, e os sensores podem já ser embutidos num vasto lote de objectos do quotidiano humano. Desta forma, tornou-se realizável fazer com que objectos inanimados, tradicionalmente não electrónicos, passem a ser inteligentes. Com a adição das capacidades de comunicação, estes passam a ter a possibilidade de interagir e cooperar com outros objectos, seja através de comunicação ad-hoc com aqueles que estão no mesmo espaço físico, ou através de uma infra-estrutura de comunicação local, em rede *wireless*, ou global, pela Internet.

Apresentam-se, de seguida, tecnologias típicas conotadas com a computação embutida que

são, usualmente, utilizadas em sistemas de Ubicomp:

- **Microprocessadores:** exemplos de processadores de sinais digitais (DSPs<sup>1</sup>), *systems-on-a-chip* (SOCs), ou microprocessadores genéricos baseados em arquitecturas VLSI/ULSI<sup>2</sup>.
- **Dispositivos de armazenamento de pequena escala:** como *chips* de memória altamente integrados, ou dispositivos de armazenamento do tipo *standalone* como cartões *compact flash*, cartões *secure digital*, e *USB memory sticks*, entre outros.
- **Comunicação ad hoc, rede e interfaces altamente integradas:** como *chips* de comunicação por radiofrequência *wireless* com suporte para Wi-Fi, Bluetooth, e ZigBee, ou portas série e paralela, ou *Universal Serial Bus* (USB), entre outros.
- **Ecrãs:** exemplos de LEDs (*Light Emitting Diodes*), ecrãs TFT (*Thin Film Transistor*), ou flexíveis orgânicos (OLEDs<sup>3</sup>).
- **Sensores:** como câmaras de vídeo miniatura, microfones, acelerómetros, sensores de pressão, ou de humidade, entre outros.
- **Actuadores:** como controlos e *switches* em miniatura, ou notificação remota baseada em comunicação, por exemplo.
- **Nós sensores:** placas remotas autónomas com capacidades de comunicação *wireless*, sensores, actuadores e microprocessador.

## 3.2 Comunicação Wireless

A interacção e a cooperação entre dispositivos computacionais distribuídos requer meios de comunicação. Esta pode ser realizada com ou sem fios. Para os dispositivos portáteis, móveis, a comunicação pode ser feita através da ligação a uma rede, quando se encontram num estado estacionário, mas na maior parte dos casos será mais adequada uma comunicação do tipo *wireless*.

Os dois meios mais populares e utilizados para a comunicação *wireless* são os infravermelhos e a radiofrequência.

### 3.2.1 Infravermelhos

A comunicação baseada em infravermelhos (IR<sup>4</sup>) pode ser considerada a precursora das tecnologias de comunicação *wireless*. Os IR são particularmente adequados para canais de co-

---

<sup>1</sup>Do anglo-saxónico *Digital Signal Processor*.

<sup>2</sup>Do anglo-saxónico *Very/Ultra Large Scale Integration*

<sup>3</sup>Do anglo-saxónico *Organic Light-Emitting Diode*.

<sup>4</sup>Do anglo-saxónico *Infra-Red*.

municação *wireless* de curta distância com uma transferência de dados entre a baixa e a média gama [68].

Os sistemas baseados em IR são simples, tendo baixos preços e providenciando uma comunicação estável sem criarem interferências consideradas prejudiciais. Esta característica torna-os adequados para a implementação em áreas sensíveis a interferências a radiofrequência, como é o caso dos telecomandos de televisão, de hospitais e de aviões [69]. A comunicação IR directa é também útil para dispositivos móveis, portáteis, uma vez que providencia uma capacidade do tipo "point & shoot".

O uso de IR permite tipicamente taxas de transmissão de até 4 Mb/s, com uma previsão para o futuro de taxas na ordem de até 100 Mb/s [70]. Contudo, o valor mais usado nos equipamentos vulgares, como portáteis e PDAs, situa-se nos 115.2 kbit/s [68].

A desvantagem relativamente à comunicação por radiofrequência situa-se no facto de suportar apenas comunicação do tipo "peer-to-peer". Mais, para conseguir as potenciais altas taxas de transferência com taxas de erro aceitáveis, o emissor e o receptor devem estar alinhados para permitirem uma comunicação em linha directa de visão.

Os dispositivos que utilizam este tipo de comunicação cumprem geralmente um padrão publicado pela *Infrared Data Association* (IrDA), daí muitas vezes os infravermelhos aparecerem com a designação IrDA, que define os padrões do protocolo para as comunicações com IR.

### 3.2.2 Radiofrequência

Com o desenvolvimento das tecnologias baseadas em radiofrequência (RF) as restrições referentes ao alinhamento e linha de visão nos IR foram ultrapassadas. Paralelamente, as taxas de transferência de dados sofreram uma alteração significativa chegando a valores numa ordem de magnitude dos Gigabit/s. A penetração das comunicações baseadas em RF foi conseguida com a introdução da família de padrões IEEE<sup>5</sup> 802.11 para as comunicações *wireless* RF, que é vulgarmente referida como *Wireless LAN* (WLAN) ou Wi-Fi (*Wireless Fidelity*). Aliás, as redes Wi-Fi baseadas na norma IEEE 802.11b, com taxas de dados até 11 Mbit/s, tornaram-se comuns em locais públicos. O desenvolvimento subsequente das comunicações *wireless* levou a taxas mais elevadas na ordem dos 54 Mbit/s (IEEE 802.11g).

A demanda por dispositivos móveis com tecnologias de comunicação *wireless* de curto alcance, mas eficientes do ponto de vista energético, levou ao desenvolvimento de tecnologias e protocolos especializados, como são os exemplos de Bluetooth e ZigBee, ambos a derivar da

---

<sup>5</sup>Do *anglo-saxónico Institute of Electrical and Eletronic Engineers*

família de normas IEEE 802.15.

No conjunto de tecnologias abordadas nesta secção foram deixadas algumas de fora que, embora interessantes do ponto de vista teórico e/ou académico, não têm expressão comercial e/ou popular, como é o exemplo da tecnologia proprietária Z-Wave [71] ou do protocolo UWB (IEEE 802.15.3a) [72], que ainda não tem uma aplicação comercial.

### 3.2.2.1 Wi-Fi

A tecnologia Wi-Fi, norma IEEE 802.11<sup>6</sup>, implementa protocolos para comunicação sem fios, projectados com o objectivo de criar redes *wireless* de alta velocidade através de ondas de rádio. O padrão Wi-Fi opera em faixas de frequências que não necessitam de licença para instalação e/ou operação, o que o torna ainda mais atractivo. A norma IEEE 802.11 utiliza técnicas de modulação OTA (*Over The Air*) que usam o mesmo protocolo básico, o CSMA/CA. De todos os protocolos que se baseiam na IEEE 802.11, os mais populares são o 802.11b e o 802.11g, que são um melhoramento ao padrão original. O 802.11a foi o primeiro padrão de rede *wireless*, mas o 802.11b foi o primeiro a ser largamente adoptado, seguido pelo 802.11g e, recentemente, pelo 802.11n [73].

É de realçar o facto de o protocolo 802.11b, que faz uso da frequência 2,4 GHz e transfere dados até uma velocidade de 11Mb/s, ter um grande sucesso tanto junto de empresas como de particulares. Esta tecnologia destina-se principalmente a computadores *desktop* e portáteis, pois por agora o seu uso em dispositivos móveis restringe-se a vários modelos de PDA's e não abrange ainda a maioria dos telemóveis, apesar de se verificarem vários casos que mostram uma franca expansão [74]. A adopção por parte de particulares tem sido enorme, sendo já muitas as habitações que usufruem de redes *wireless* Wi-Fi. Por outro lado, actualmente, o uso da tecnologia segue o objectivo das empresas de pretenderem diminuir a quantidade de ligações físicas para a realização das comunicações. Em geral, para a implementação de uma rede Wi-Fi, deverá ser feita a instalação de *hardware* de recepção de sinais de rádio e de pontos de acesso, para a emissão e recepção dos sinais [73].

Uma das principais desvantagens do Wi-Fi em relação a outras tecnologias *wireless* é o elevado gasto de energia. Ou seja, para cobrir uma grande área de rede, é necessário um alto consumo de energia, já que o alcance do Wi-Fi está directamente relacionado com a potência do transmissor dos sinais de rádio [73].

---

<sup>6</sup><http://www.ieee802.org/11/>

Protocolo	Ano de Origem	Taxa Dados OTA (Est.)	Taxa Dados MAC SAP (Est.)	Modulação	Alcance (indoor)
802.11a	1999	54 Mbps	25 Mbps	OFDM	35 m
802.11b	1999	11 Mbps	5 Mbps	DSSS	38 m
802.11g	2003	54 Mbps	25 Mbps	OFDM	38 m
802.11n	2007	até 600 Mbps	até 400 Mbps	MIMO	70 m

Tabela 3.1: IEEE 802.11: Características dos seus protocolos para transferência de dados.

### 3.2.2.2 Bluetooth

O Bluetooth (BT) [6] é uma norma (IEE 802.15.1) de baixo custo, baixa complexidade e pouca potência, cobrindo distâncias curtas. Tendo sido, inicialmente, projectado para substituir os cabos de curta distância é possível ligar facilmente vários tipos de dispositivos de comunicação, tais como PCs, *notebooks*, *palmtops*, *handhelds*, impressoras, telemóveis, enfim, qualquer dispositivo que possua um *chip* BT. Esta tecnologia já se encontra presente em muitos portáteis, telemóveis e PDA's, e pode ser adicionada através da utilização de um adaptador USB (ver Figura 3.1) a todos os dispositivos que suportem esta ligação e que não possuam BT de origem. Actualmente, esta tecnologia encontra-se em muitos dispositivos, permitindo uma comunicação rápida e eficiente entre dispositivos e apresentando uma vasta gama de aplicações.



Figura 3.1: Comunicação Bluetooth através do adaptador USB.

Os dispositivos BT trabalham na faixa de frequências ISM (*Industrial, Scientific, Medical*), em 2,45 GHz. A comunicação entre eles é feita através de um canal FH-CDMA (*Frequency Hopping - Code-Division Multiple Access*). Numa comunicação BT, os dispositivos comunicam entre si formando uma pequena rede *wireless* denominada por Piconet, na qual podem existir até oito dispositivos interligados, sendo um deles o mestre e os outros escravos. Todos os dispositivos

### 3.2. COMUNICAÇÃO WIRELESS

mantêm uma ligação lógica com os restantes. Essa ligação lógica apenas é convertida numa ligação física quando existe a necessidade de troca de dados entre os dispositivos. Os componentes da rede BT são identificados através de um endereço único, de uma forma bastante parecida com o endereço MAC numa rede Ethernet. Esse endereçamento e identificação são usados para regular a autorização, as funcionalidades e a segurança dos vários participantes. As redes BT inserem-se na denominação de rede de comunicação pessoal (WPAN).

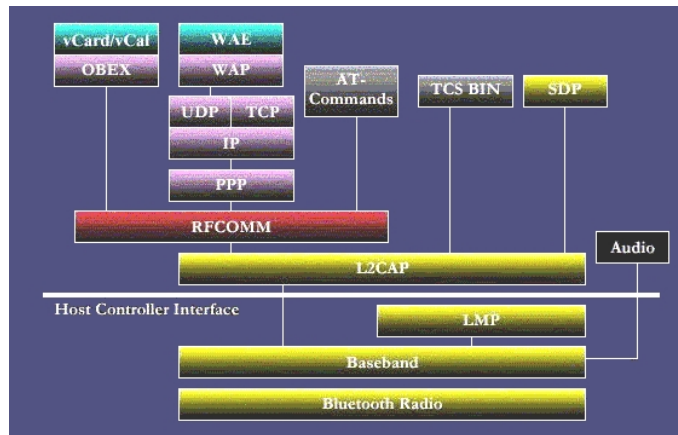


Figura 3.2: Bluetooth: Arquitectura completa da pilha de protocolos [6].

A arquitectura do BT e as suas características técnicas estão definidas nas especificações denominadas *Core* (Núcleo) e *Profiles* (Perfis), definidas pelo SIG [75]. Enquanto a especificação do núcleo define como o sistema funciona (protocolos, camadas, especificações técnicas, etc.), o documento que define os perfis determina como os diversos elementos que compõem o sistema podem ser usados na realização das funções desejadas. Na arquitectura do BT (ver Figura 3.2) podem ser encontrados os seguintes protocolos que fazem parte do *core* da especificação BT:

- *Baseband*: Em conjunto com a LCL (*Link Control Layer*) permite a ligação de RF dos dispositivos numa piconet. Esta camada utiliza procedimentos de inquirição e *paging* para sincronizar a transmissão entre os diversos dispositivos BT.
- LMP (*Link Manager Protocol*): Protocolo responsável pela preparação de canais de ligação entre dispositivos após executar métodos de segurança, tais como autenticação e encriptação.
- L2CAP (*Logical Link Control and Adaptation Protocol*): Os pacotes L2CAP carregam *payloads* que são levados para os protocolos de camadas superiores. Protocolo de adaptação e controlo do enlace lógico.
- SDP (*Service Discovery Protocol*): Com o uso do SDP, a informação sobre o dispositivo,

os serviços permitidos e as características dos serviços são questionados entre dispositivos BT activados. É usado para encontrar serviços em dispositivos remotos de BT.

- RFCOMM (*Cable Replacement Protocol*): Protocolo virtual do porto de série para a comunicação.
- PPP, TCP/UDP/IP, OBEX e WAP: Protocolos suportados pelo BT para maximizar a interoperabilidade.

Uma vez que a banda ISM que o BT utiliza não possui licenças, apesar de regulamentada, convergiu-se para um único padrão. Deste modo, é possível utilizar os dispositivos em qualquer local, não sendo necessária uma norma legal para usar a tecnologia. Tendo em consideração o elevado número de dispositivos móveis, e pessoais, que actualmente implementam a tecnologia, esta tem realmente uma característica ubíqua que a torna relevante no momento de análise para desenvolvimento de qualquer sistema de UbiComp.

### 3.2.2.3 Comparação das Normas IEEE 802

A Figura 3.3 apresenta um quadro comparativo ao nível das principais características das três normas com mais afirmação no momento, Wi-Fi, Bluetooth e ZigBee.

Feature(s)	IEEE 802.11b	Bluetooth	ZigBee
Power Profile	Hours	Days	Years
Complexity	Very Complex	Complex	Simple
Nodes/Master	32	7	64000
Latency	Enumeration up to 3 Seconds	Enumeration up to 10 seconds	Enumeration 30ms
Range	100 m	10m	70m-300m
Extendibility	Roaming Possible	No	YES
Data Rate	11Mbps	1 Mbps	250Kbps
Security	Authentication Service Set ID (SSID), WEP	64 bit, 128 bit	128 bit AES and Application Layer user defined

Figura 3.3: Quadro comparativo de protocolos de radiofrequência [6].

O Zigbee [76], baseado na norma IEEE 802.15.4, foi apresentado em 2005 pela ZigBee Alliance<sup>7</sup>, entidade responsável pela sua manutenção. As áreas de aplicação do ZigBee vão desde o controlo industrial à domótica, possuindo características importantes como, por exemplo, um reduzido consumo de energia, uma elevada fiabilidade, e um menor tempo de ligação à rede que os outros protocolos, apresentando maior rapidez na passagem do modo *standby* a activo.

<sup>7</sup><http://www.zigbee.org/>

Contudo, embora seja muito promissor, a falta de maturidade de ferramentas disponíveis no mercado dificulta, por agora, a utilização alargada e quotidiana deste protocolo [77].

As três tecnologias não concorrem entre si. De facto, o ZigBee foi desenvolvido com o objectivo de realizar monitorização de sistemas, enquanto que o BT é mais apropriado para aplicações que requerem um maior débito sobre pequenas redes, como por exemplo as redes ad-hoc e os sistemas para transmissão de áudio ou de dados *peer-to-peer*. O ZigBee é mais adequado para transmitir pacotes menores (e.g., de texto) sobre grandes redes. Verifica-se que é ideal para aplicações com dispositivos remotos alimentados por baterias, nomeadamente os sensores e os actuadores, já que permite baixos consumos, débitos aceitáveis e possui uma pilha protocolar mais simples que possibilita a sua implementação em sistemas com recursos limitados. O Bluetooth pode ser visto como uma substituição de cabos em produtos como computadores portáteis, telemóveis e auriculares, funcionando como uma espécie de USB *Wireless*.

Quanto ao Wi-Fi, este tem sido popularizado pela grande capacidade observada na ligação à Internet, e pelas redes de computadores, mormente as domésticas. Já é indicado para grandes transferências de ficheiros, com conteúdos mais pesados, e para vídeo.

### 3.2.2.4 NFC (*Near Field Communication*)

A NFC [78] é uma tecnologia recente de comunicação bidireccional *wireless* baseada em RFID, sendo por vezes chamada de tecnologia *contactless*. A tecnologia NFC<sup>8</sup> é uma comunicação do tipo *peer-to-peer*, que trabalha na frequência dos 13,56 MHz e apresenta transferências de dados a uma velocidade até 424 Kbps. Sendo necessário aproximar os dispositivos a uma distância de até quatro centímetros para iniciar uma comunicação, esta tecnologia também é compatível com outras tecnologias *wireless*, como Bluetooth ou Wi-Fi. São vários os dispositivos que podem usar NFC, desde telemóveis (ver a Subsecção 3.6), que permitem pagamentos ou transferência de informação, a câmaras digitais, que enviam as suas fotos para um ecrã através de um contacto. As possibilidades de uso para a tecnologia são muitas, como por exemplo em superfícies comerciais onde um telemóvel equipado com NFC pode ser utilizado para efectuar os pagamentos como se de um cartão de crédito se tratasse. Para tal, serão incluídas medidas de segurança de forma a proteger os dados financeiros e assegurar transacções seguras aos consumidores. São quatro as categorias essenciais nas quais se podem dividir as aplicações NFC [11]: "*Touch and Go*"; "*Touch and Confirm*"; "*Touch and Connect*"; e "*Touch and Explore*".

As grandes vantagens da tecnologia, que no futuro permitirá que telemóveis com suporte

---

<sup>8</sup>A NFC foi inicialmente desenvolvida pela Philips e pela Sony.

para NFC possam trocar dados pessoais, prendem-se com a facilidade de configuração. Basta aproximar dois dispositivos que estes iniciam automaticamente as ligações de rede, sem necessitar de qualquer tipo de configuração por parte do utilizador. Os dispositivos podem comunicar em modo activo ou passivo. No primeiro dos modos, ambos geram campos de RF para transportar os dados. No modo passivo, apenas um dispositivo emite o campo de RF, enquanto que o outro transfere os seus dados por modulação.

A NFC pode ser facilmente comparada com a BT, uma vez que são tecnologias de curto alcance que têm vindo a ser adoptadas pelos fabricantes de telemóveis, com destaque para a BT que é a mais antiga. A NFC apresenta como grande vantagem o facto de necessitar de menos tempo para a iniciação da comunicação, não necessitando de uma configuração manual como acontece com a BT. Para evitar casos de complicadas configurações, a NFC até pode ser usada para fazer a iniciação de outras tecnologias, como a própria BT. Por outro lado, enquanto que a taxa máxima de transferência da NFC se situa nos 424 kbps, a da BT já vai até aos 3Mbps. Contudo, o alcance muito reduzido da NFC oferece um maior grau de segurança, fazendo com que seja adequada para ambientes de multidões e para cenários onde o utilizador se deve sentir na posse do controlo do processo. A NFC apresenta uma outra vantagem ao ser compatível com estruturas RFID existentes, enquanto que a BT será adequada para a detecção de utilizadores sem que estes tenham que estar muito próximos do dispositivo de leitura.

### 3.3 Identificação e Localização

Como regra, a coordenação da interacção entre entidades distribuídas requer de alguma forma um meio de discriminação. Ou seja, é extremamente importante garantir que uma única entidade seja correctamente identificada, i.e., obter uma identificação para que se lhe possa atribuir, de forma não ambígua, responsabilidades, funcionalidades e dados. Este cenário assemelha-se ao dos dispositivos em rede, que necessitam de possuir um identificador de endereço para que se faça a comunicação *peer-to-peer*.

Dependendo do ambiente que se queira implementar, são vários os meios tecnológicos que podem ser utilizados. As próximas subsecções apresentam aqueles que são mais utilizados e que correspondem a tecnologias mais comuns.

#### 3.3.1 Identificadores de Endereço Único

Um meio de identificação natural passa pelos identificadores de endereço único usados em protocolos de rede e comunicação. São exemplos de identificadores os endereços lógicos IP e os endereços MAC físicos utilizados enquanto interfaces de comunicação de *hardware*, e como os encontrados em sistemas em rede usando *Ethernet* (Wi-Fi) ou *Bluetooth*, por exemplo. Por outro lado, verifica-se que a NFC é uma tecnologia promissora em termos de identificação exactamente por ser uma comunicação de curto alcance que apresenta uma configuração fácil e rápida.

O identificador de endereço de um objecto pode ser adquirido caso a entidade inquiridora tenha a mesma interface de comunicação, tendo que ser possível estabelecer um canal de comunicação entre ambos, seja directamente via ad hoc ou indirectamente via conectividade rede. Assim sendo, num sistema de UbiComp, desde que, por exemplo, as entidades estejam ligadas em rede, com ou sem fios, podem ser unívocamente identificadas.

#### 3.3.2 Visual Tags

As *Visual Tags*, ou *Visual codes*, podem ser associadas a objectos ou até impressas, necessitando de uma linha de visão desobstruída para serem directamente lidas. Alguns exemplos de *tags* a destacar são as vulgares barras de código unidimensionais, lidas por meio de um *scanner* de infravermelhos, ou as barras de código 2D, com exemplos a serem os sistemas CyberCode [7] da Sony e o VisualCodes [8] do ETH (Instituto Federal Suíço da Tecnologia). As *visual tags* 2D podem ser lidas por leitores específicos para o efeito ou por vulgares câmaras CCD que a maior parte dos telemóveis já integram [7, 8]. O ARToolKit [9], uma das plataformas para Realidade Aumentada mais utilizadas, usa técnicas de visão computacional para calcular a posição e a orientação de uma câmara em relação a *visual tags*, identificando deste modo onde sobrepôr, em tempo real, os objectos virtuais (ver Figura 3.5).



Figura 3.4: Exemplos de *visual tags* da CyberCode [7].

As *tags* consistem usualmente em padrões brancos e pretos que codificam o ID, e por vezes meta-informação adicional que permite que o dispositivo de leitura possa determinar o alinhamento e a distância relativamente à *tag*. Os padrões típicos são as riscas no caso das barras de códigos e quadrados brancos e pretos bidimensionais para as *visual tags*. Regra geral, quando aparece o termo *visual tags* na bibliografia este é referente às barras 2D, que são mais comuns em sistemas de UbiComp.

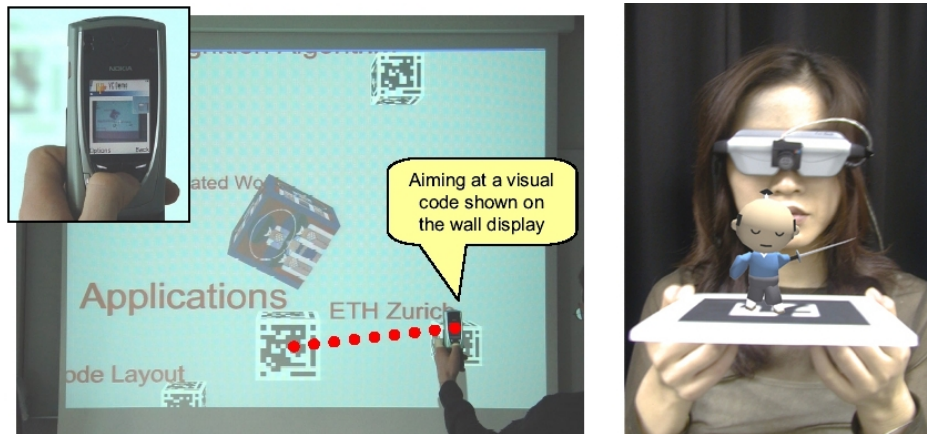


Figura 3.5: (esquerda) ETH VisualCodes: Telemóvel a fotografar uma *visual code* [8]; (direita) ARToolkit: Visualização de elemento virtual sobre uma *visual tag* [9].

As *visual tags* têm como vantagem, por exemplo, relativamente aos identificadores de endereço referidos no ponto anterior, o facto de os objectos identificados por elas não necessitarem de possuir capacidades de computação ou comunicação pois apenas necessitam de apresentar a *tag*. A implementação deste método de identificação num ambiente torna-se mais suave e adapta-se mais facilmente à generalidade dos objectos. Por outro lado, as entidades que fazem a identificação já necessitam de ser ou ter leitores próprios para o efeito, e mesmo que sejam utilizadas as câmaras de telemóvel, estes devem possuir aplicações de processamento de imagem que façam o reconhecimento das *tags*.

### 3.3.3 RFID (*Radio Frequency Identification*)

A tecnologia RFID tem vindo a substituir as barras de códigos em muitos cenários industriais de identificação e rastreio de produtos. Esta tecnologia, de identificação por radiofrequência, tem crescido recentemente fruto das imensas aplicações que têm vindo a surgir nas mais diversas áreas [11].

Existem vários métodos para identificar objectos através de RFID, mas o mais comum passa

### 3.3. IDENTIFICAÇÃO E LOCALIZAÇÃO

---

por guardar um ID ou número de série que identifica um produto específico, juntamente com outra informação, numa etiqueta (*tag*), que é composta por um pequeno *microchip* anexado a uma antena, e que pode ser colocada numa pessoa, animal ou objecto. A antena permite que o *chip* transmita a informação de identificação a um leitor. O leitor converte as ondas rádio da *tag* RFID em informação digital que poderá ser processada pelos sistemas de informação.

Tipicamente, quando um leitor lê uma *tag* passa para um computador, designado por *host* do sistema, três parâmetros elementares: o ID da *tag*, o ID do próprio leitor, e o instante temporal em que a *tag* foi lida/detectada. Conhecendo as localizações dos leitores, um sistema RFID pode saber onde se encontra um determinado item, e rastreando a *tag* em termos temporais, saber o percurso que fez.

A implementação de uma infra-estrutura RFID completa baseia-se numa composição de quatro componentes (ver Figura 3.6):

- *tags*, também conhecidas por *transponders* ou cartões de proximidade, que podem ser encontradas sob diferentes formatos: passivas, semi-passivas ou activas. As suas dimensões e aplicações variam consoante o formato.
- leitores, também conhecidos por *transceivers*, em conjunto com as suas antenas.
- um pacote *middleware*, que facilita a comunicação entre o *hardware* e a aplicação, além de aumentar a precisão dos dados.
- aplicações de *software* para a gestão dos dados.

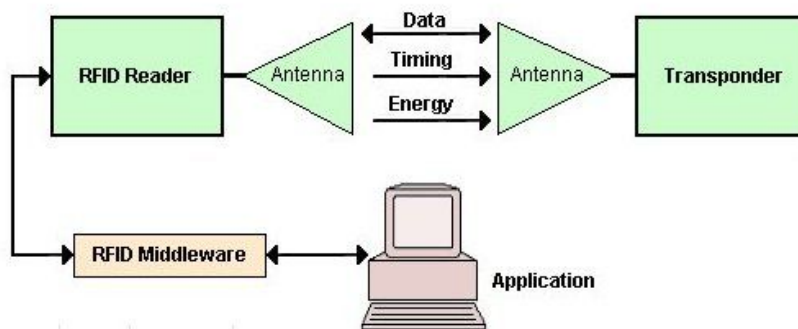


Figura 3.6: Infra-estrutura típica de um sistema RFID.

As *tags* passivas não possuem qualquer tipo de alimentação, passando a maior parte do tempo "adormecidas", sendo activadas quando se encontram numa zona de leitura (conjunto antena ligada a um leitor). Não requerem qualquer tipo de manutenção e possuem tamanhos cada vez menores, devido à ausência de bateria, chegando actualmente a terem dimensões na

ordem da décima de milímetro, e na ordem dos micrómetros de espessura, mesmo tendo em conta o tamanho e desenho da antena embutida nas *tags* [11]. Por outro lado, as *tags* activas têm um tamanho superior ao das passivas, uma vez que possuem uma bateria e um circuito rádio que lhes permite transmitir o próprio sinal para o leitor, não dependendo deste para serem alimentadas. Oferecem um alcance muito superior, mas são mais pesadas e financeiramente mais dispendiosas. Estas *tags* são ideais para uso em sistemas de RTLS, que serão apresentados na Subsecção 3.3.5.

A quantidade de informação numa *tag* pode variar entre apenas alguns bytes e vários Megabytes de informação. Existem muitos formatos proprietários para a organização da informação, mas o *Electronic Product Code* (EPC) é considerado a norma RFID. O novo EPC usa o *General Identifier* (GID-96) da organização EPCglobal<sup>9</sup>, contendo 96 bits que são divididos da seguinte forma: 8 para o cabeçalho, 28 para a empresa, 24 para a classe do objecto e 36 para o número série.



Figura 3.7: RFID: (cima) Antenas e (baixo) leitores (fixo e móvel).

Os leitores são usados para ler, interpretar e escrever *tags*, ligando-se a uma ou mais antenas, por intermédio de uma qualquer interface (os cabos coaxiais são a mais comum) definida pelo construtor. São vários os tipos de leitores que podem ser encontrados, dependendo de características como, por exemplo, a frequência de trabalho, o tipo de mobilidade, ou as interfaces de comunicação. Para a escolha do leitor, a característica da mobilidade é muito importante, encontrando-se intimamente ligada ao tipo de sistema que se pretende implementar. Os leitores podem ser fixos, móveis ou manuais, podendo ser usados em cenários como, por exemplo, a entrada de um armazém, ou ser carregados por funcionários. Quanto às antenas, estas são normalmente alimentadas pelo próprio leitor, mas podem ter alimentação própria. Existem vários

<sup>9</sup><http://www.epcglobalinc.org/>

### 3.3. IDENTIFICAÇÃO E LOCALIZAÇÃO

---

tipos de antenas (ver Figura 3.7), sendo as antenas de *desktop* ou de parede as mais comuns. Estas podem ser agrupadas para funcionarem como antenas de portal [11].

Relativamente ao *middleware*, no contexto específico da tecnologia RFID esta camada deve executar funções adicionais, tais como: fazer com que a leitura/escrita seja mais confiável; fazer circular os dados numa rede de leitores para a localização correcta; controlar e monitorizar os leitores; permitir operações seguras de leitura/escrita; reduzir as interferências de RF; e notificar as aplicações.

Segundo a Venture Development Corporation<sup>10</sup> (VDC) [79], o *middleware* de RFID tem evoluído muito, oferecendo maiores flexibilidade e escalabilidade, para as implementações da tecnologia, e simplificando a integração de sistemas de RFID com outras plataformas. São várias as propostas de *middleware* que podem ser encontradas. Tanto surgem de empresas dedicadas ao seu desenvolvimento, como são os casos da OATSystems<sup>11</sup> e da GlobeRanger<sup>12</sup>, como aparecem no seio de grandes empresas que pretendem alargar o conjunto de soluções que propõem (e.g., IBM, Oracle ou SAP), ou em empresas dedicadas à implementação de redes de sensores e identificação (e.g., Blue Vector<sup>13</sup>), entre outras.

#### 3.3.4 GPS (*Global Positioning System*)

O *Global Positioning System* (GPS), ou Sistema de Posicionamento Global, é um sistema de posicionamento por satélite utilizado para determinação da posição de um receptor na superfície da Terra ou em órbita, sendo por vezes incorrectamente designado de sistema de navegação. Este sistema<sup>14</sup> foi criado e é controlado pelo Departamento de Defesa dos Estados Unidos da América (DoD), e pode ser utilizado por qualquer pessoa, gratuitamente, necessitando apenas de um receptor que capte o sinal emitido pelos satélites [80]. Actualmente, existem dois sistemas alternativos ao GPS: o GLONASS<sup>15</sup>, pertencente à Federação Russa; e o GALILEO<sup>16</sup>, um projecto da União Europeia com a Agência Espacial Europeia ESA.

O DoD fornece dois tipos de serviços: o *Standard Positioning Service* (SPS) e o *Precise Positioning Service* (PPS). O PPS consegue processar paralelamente os dados adquiridos pelos satélites visíveis, enquanto o SPS usa um processamento multiplexado, i.e., recebe dados de pelo

---

<sup>10</sup><http://www.vdc-corp.com/>

<sup>11</sup><http://www.oatsystems.com/>

<sup>12</sup><http://www.globeranger.com/>

<sup>13</sup><http://www.bluevector.com>

<sup>14</sup>Foi-lhe dado o nome oficial NAVSTAR GPS.

<sup>15</sup><http://www.glonass-ianc.rsa.ru/>

<sup>16</sup>[http://ec.europa.eu/dgs/energy\\_transport/galileo/index\\_en.htm](http://ec.europa.eu/dgs/energy_transport/galileo/index_en.htm)

menos três satélites visíveis, mas um de cada vez [80]. Inicialmente, a precisão do SPS era na ordem dos 100 metros e a do PPS chegava aos 20 metros. O serviço SPS tinha uma taxa de erro superior pois continha um erro artificial adicionado pelo DoD, alegadamente por razões de segurança. Em Maio de 2000, esse erro deixou oficialmente de ser introduzido passando a ter uma taxa de erro semelhante à do outro serviço [53].

O GPS necessita de obter os dados de, pelo menos, 4 satélites com um sinal suficientemente estável. Torna-se impraticável utilizar este sistema em locais com pouca cobertura como, por exemplo, no interior dos edifícios, ou com vegetação densa. Isto leva ao efeito "*urban canyon*" nas cidades, onde a funcionalidade GPS é perdida devido à ocultação ou à dispersão, em várias direcções, dos sinais por parte dos edifícios [81]. Como resultado, o GPS funciona bem nos exteriores, mas a cobertura nos interiores, onde as pessoas passam a maior parte do tempo, é nenhuma ou muito reduzida. Sendo assim, tem sido pouco visto em projectos e implementações na área da Ubicomp [81]. Uma possível solução para o efeito "*urban canyon*" poderia ser o uso de *repeaters* GPS colocados nos extremos dos edifícios para retransmitirem os sinais para o interior [82].

O GPS é, talvez, o exemplo mais familiar de um sistema de localização que providencia uma cobertura a larga escala, i.e., mundial, tornando-o teoricamente ubíquo. Contudo, e apesar desta vantagem, o sistema contém a restrição dos interiores que não permite que o mesmo possa ser considerado verdadeiramente ubíquo. No entanto, o GPS pode tornar-se uma característica comum a quase todos os aspectos da vida humana do futuro.

#### 3.3.5 RTLS (*Real Time Locating Systems*)

Os sistemas de localização em tempo real (RTLS) [83] são usados para seguir e identificar objectos em tempo real, usando para tal nós activos (*badges/tags*), pouco dispendiosos e simples, anexados ou embutidos nos objectos, e dispositivos de leitura (*readers*) que recebem sinais *wireless* a partir dos nós para determinarem as suas localizações. De entre os cenários onde a tecnologia RTLS pode ser útil destacam-se os seguintes: ferramentas e utensílios caros que não podem ser achados facilmente quando são necessários; linhas de produção complexas onde itens não são localizados facilmente; ou complexos que necessitam seguir o trânsito de pessoas dentro e fora do perímetro.

Uma solução RTLS utiliza, tipicamente, *tags* rádio com bateria e um sistema de localização celular. O sistema de localização é normalmente implementado como uma matriz para os dispositivos de leitura que são instalados com um espaçamento algures entre os 50 e os 1000 pés. Os

### 3.3. IDENTIFICAÇÃO E LOCALIZAÇÃO

---

sistemas fazem uma actualização contínua de uma base de dados com as localizações correntes das *tags*. A frequência de actualização pode ser na ordem dos segundos ou das horas, para o caso de itens que raramente se movem. Um sistema poderá fazer o rastreamento simultâneo de milhares de *tags* e as baterias das mesmas podem durar mais do que a referência de 5 anos.

São várias as tecnologias que podem ser usadas em sistemas de localização em tempo real, podendo destacar-se as seguintes: RFID Activo; Híbrido RFID Activo - IR; Infravermelhos; *Radio Beacon*; UWB; e Wi-Fi, entre outras. Alguns dos casos de sucesso encontram-se em empresas como: Kimaldi<sup>17</sup> ou RTLS Inc<sup>18</sup> (com RFID activo); PINCSolutions<sup>19</sup> ou Ekahau<sup>20</sup> (com Wi-Fi); e UbiSense<sup>21</sup> (com UWB). O sistema GPS também pode ser visto como RTLS, mas não será apropriado para o rastreamento de muitos objectos pela baixa precisão que apresenta e por ser essencialmente indicado para espaços exteriores.



Figura 3.8: *Tags* para RTLS: (esquerda) UWB, (centro) Wi-Fi e (direita) RFID Activo.

Um tipo particular de RTLS são os sistemas LLS (*Local Locating System*) para o rastreamento de objectos dentro de áreas restritas. As aplicações mais comuns da tecnologia RFID, utilizando *tags* passivas, são um bom exemplo de LLS. Os leitores LLS são instalados na área a rastrear, e são eles que comunicam, periodicamente, com todas as *tags* para confirmar que estas se encontram no seu raio de acção, assim estimando a localização dos objectos. As *tags* usadas em RTLS são lidas automaticamente e continuamente, estando em comunicação com os leitores, independentemente do processo que as move. Com os RTLS, não será necessária a intervenção humana ou processos controlados para determinar a localização dos objectos. Por outro lado, com os sistemas RFID não activos não se garante a localização corrente de um objecto, mas a localização onde foi "visto" pela última vez.

---

<sup>17</sup><http://www.kimaldi.com/>

<sup>18</sup><http://www.rtls.com/>

<sup>19</sup><http://www.pincolutions.com/>

<sup>20</sup><http://www.ekahau.com/>

<sup>21</sup><http://www.ubisense.net/>

## 3.4 Sensores

Uma preocupação fundamental para a computação baseada em informação contextual é a noção do espaço físico que envolve os objectos, um utilizador e/ou o seu dispositivo móvel. Com os avanços que se têm registado ao nível da tecnologia de sensores, o conhecimento das condições físicas do ambiente pode agora ser um factor de decisão embutido em objectos e dispositivos móveis a baixo custo.

Os sensores a serem considerados num ambiente de Ubicomp devem ser pequenos (o máximo possível), baratos (para serem utilizados em grandes quantidades), pouco consumidores de energia, capturando parâmetros físicos a partir do ambiente ou de objectos, e actuando como entrada para funções que determinem a sua informação contextual geral.

Segundo um estudo de 2004 realizado por Beigl et al. [84], no qual foram analisados mais de 300 nós sensores, são sete as classes gerais nas quais se podem agrupar os sensores, de acordo com parâmetros físicos que providenciam informação útil aos objectos:

- Movimento - considerados os tipos de movimento: "a ser usado", "correndo", "permanecendo parado", além da aceleração, rotação e vibração;
- Força/Pressão - inclui medições de peso e força a várias partes de um objecto;
- Luz - vários comprimentos de onda como a luz de dia ou luz de infra-vermelhos, incluindo parâmetros como o nível da luz e as alterações ao longo do tempo;
- Temperatura - em várias zonas de um objecto medindo a temperatura do ambiente, temperatura do objecto, ou de um outro objecto em "interacção";
- Humidade - em vários locais como humidade do ambiente ou humidade no interior do objecto;
- Áudio - nível de ruído, espectro de frequência, mas também alterações ao longo do tempo;
- Proximidade - detecção de actividade próxima no ambiente.

Ainda segundo o mesmo estudo, os sensores mais utilizados em aplicações de Ubicomp são os de movimento, luz e força/pressão, que são especialmente indicados para actividades de reconhecimento no ambiente e no objecto ao qual o sensor se encontra ligado. Existem outras duas categorias de sensores, temperatura e áudio, que também estão muito presentes em nós de redes sensores [84]. A placa sensorial genérica do projecto Particle<sup>22</sup> é um bom exemplo de utilização de todos os tipos de sensores, exceptuando o de proximidade (ver Secção 3.5).

---

<sup>22</sup><http://particle.teco.edu/>

#### 3.4.1 Redes de Sensores

Uma Rede de Sensores Sem Fios (RSSF ou WSN<sup>23</sup>), ou apenas rede de sensores, consiste num determinado número de sensores espalhados por uma zona geográfica.

Os recentes avanços nas tecnologias das comunicações *wireless* e dos *microchips* são os principais responsáveis pelo desenvolvimento de nós sensores multifuncionais, de baixo custo e baixo consumo energético, que são pequenos em tamanho e conseguem comunicar sem fios em curtas distâncias. Regra geral, consistem em componentes de monitorização, processamento de dados, e comunicação. Com a implementação de um grande número de nós sensores no interior ou junto a um fenómeno a ser observado obtém-se a rede de sensores. Estas redes permitem acompanhar, monitorizar, estudar, compreender e actuar sobre o fenómeno pretendido.

O cenário óptimo para a instalação de uma RSSF seria um ambiente onde não existissem interferências nas comunicações, onde o terreno fosse plano e não existisse vegetação [85]. Num cenário ideal, a melhor solução para a disposição dos seus elementos é em forma de grelha, onde os sensores são colocados lado a lado a uma distância  $d$ , sendo este parâmetro proporcional ao alcance rádio de cada um dos nós [86]. Contudo, a colocação dos sensores é muitas vezes realizada em ambientes inhóspitos. Geralmente, a posição dos nós sensores não é predeterminada, permitindo assim uma implementação aleatória para operações em terrenos inacessíveis ou de socorro a acidentes.

Uma característica única nestas redes é o suporte à comunicação entre os seus nós. Uma tarefa típica de um nó é a fusão de dados recebidos de outros nós. Uma vez que os nós sensores são equipados com um microprocessador, ficam habilitados a realizar processamentos e, assim, a transmitir apenas os dados requisitados, e parcialmente processados, para outros nós, em vez de enviarem os dados em bruto. Deste modo, a rede consegue reduzir globalmente os gastos energéticos e o tempo de comunicação.

As aplicações de RSSF requerem, usualmente, técnicas de rede do tipo *wireless* ad-hoc para a coordenação de actividades cooperativas. No entanto, Akyildiz et al. referem que os protocolos e algoritmos propostos para as tradicionais redes *wireless* ad-hoc não se ajustam por completo aos requisitos das aplicações de RSSF, apresentando uma lista de diferenças [87].

Existe ainda a possibilidade de os nós sensores interagirem com PDAs, ou até telemóveis, possivelmente através de protocolos como, por exemplo, o Bluetooth, que já equipa muitos dos dispositivos móveis [88]. Esta interacção permitirá que muitas das leituras feitas pelos objectos inteligentes possam ser comunicadas automaticamente para as entidades móveis do ambiente.

---

<sup>23</sup>Do anglo-saxónico *Wireless Sensor Network*.

Apesar de esta comunicação não ser de fácil implementação [89], têm aparecido casos de sucesso de comunicação com Bluetooth entre plataformas de nós sensores e PDAs [90].

### 3.5 Plataformas de Desenvolvimento

O desenvolvimento de "novos" dispositivos e objectos inteligentes para aplicações específicas de UbiComp é uma tarefa reconhecidamente difícil. O problema maior encontra-se na dificuldade em desenvolver e combinar dispositivos físicos e em disponibilizar interfaces para as linguagens de programação convencionais [91].

Têm surgido várias plataformas de desenvolvimento com vista a auxiliar os investigadores na criação de objectos e dispositivos inteligentes. Por exemplo, Greenberg e Fitchett resolveram desenvolver de raiz um *toolkit* chamado **phidgets**<sup>24</sup> [91]. Este *toolkit* faz a abstracção e a integração de dispositivos de entrada e de saída, escondendo os detalhes de implementação, oferecendo uma API (*Application Programming Interface*) de funcionalidades, e apresentando uma interface interactiva (opcional) para a visualização e controlo do estado do dispositivo.

De seguida, apresentam-se algumas das plataformas mais relevantes, com destaque para a área das redes de sensores, que facilitam o desenvolvimento de ambientes inteligentes [92]:

- **BTNodes** [93]: O BTnode é uma plataforma autónoma de computação e comunicação *wireless* que é baseada em *Bluetooth* e num microcontrolador. Serve enquanto plataforma de demonstração para a investigação em redes móveis e ad-hoc, as MANETs (*Mobile and Ad-hoc coNnected Networks*), e em redes de sensores distribuídas. Esta plataforma pode estar temporariamente ligada a uma infra-estrutura *backend* para o armazenamento dos dados e para interacção com o utilizador, podendo ainda usar actuadores e dispositivos de computação com recursos para tarefas complexas de processamento de sinais.
- **Motes** [94]: O projecto Motes vai na versão **Imote2** (IPR2400), uma plataforma nó sensor *wireless* que permite o desenvolvimento de sistemas poderosos, automatizados e sem fios, para a monitorização e recolha de dados. É construída à volta do processador PXA271 XScale da Marvell, integrando comunicação RF Zigbee com uma antena nativa de 2.4GHz. A Imote2 é modular, podendo ser expandida com placas de extensão para adequar o sistema a aplicações específicas.

---

<sup>24</sup>De *physical widgets*, sendo análogo ao termo *widget* que designa os componentes de interface gráfica (e.g., janelas e botões) com o utilizador (GUI).

- **SoapBox** [95]: A SoapBox (*Sensing, Operating and Activating Peripheral Box*) foi desenvolvida enquanto ferramenta de investigação, e para ser utilizada como protótipo genérico para a experimentação de ideias de produtos e desenvolvimento dos respectivos protótipos. Enquanto plataforma configurável, a SoapBox oferece características elementares como um muito baixo consumo energético, sensores nativos e tamanho reduzido. Contudo, apesar de apresentar comunicação *wireless* é a mais limitada neste ponto.
- **Smart-Its** [96]: O projecto Smart-Its apresenta uma plataforma direccionada para os sistemas de *context-aware* e aumento de objectos, sendo uma das mais referenciadas na literatura. As placas Smart-Its são pequenas, integrando as habituais capacidades de sensorização, actuação, processamento e comunicação. Existem em duas versões que implementam a mesma arquitectura base: uma chamada **Lancaster Smart-Its** que é desenhada para a prototipagem rápida; e outra conhecida por **TecO Smart-Its** que é mais adequada para a implementação de sistemas completos. A primeira é uma *toolbox* extensível de módulos de *hardware* projectada para a prototipagem de objectos e artefactos aumentados, tendo sido desenvolvida como plataforma experimental para o suporte de interacções entre as componentes digital e física de um objecto. Podem ser adaptadas para cada aplicação, onde o *hardware* permite uma grande flexibilidade de configurações.
- **TecO Smart-Its Particle** [97]: A plataforma Particle pode ser vista como sendo a sucessora da Smart-Its. A placa Particle tem sido amplamente utilizada nos meios académico e industrial numa escala bem pequena (19x35 mm) [84]. As Particles são desenhadas para serem mais pequenas e para consumirem menos energia com vista a manterem-se em acção por muito tempo (e.g., várias semanas) sem necessitarem de manutenção. Esta plataforma dirige-se para a prototipagem rápida, sendo indicada para redes de sensores, computadores "wearable", automação caseira, e espaços inteligentes. Além de integrar os mais variados sensores, existe ainda uma unidade de processamento embutida e uma interface para outras placas sensoriais ou para placas de comunicação *wireless* [84]. Apresenta um conjunto de APIs que é integrado no sistema operativo que corre na unidade de processamento.

Exemplos de outras plataformas são a iBeans [98], a Cube sensor [99], a PadNET [100], ou a Ember [101].

## 3.6 Dispositivos Móveis Pessoais

Os dispositivos pessoais, como os telemóveis e os PDAs, desempenham um papel importante na construção de muitos dos sistemas de Ubicomp [102, 103, 104]. A sua utilização generalizada e as suas características, enquanto plataforma de computação com um propósito geral, fazem deles dispositivos ideais na implementação de muitos serviços interactivos em ambientes inteligentes. Cenários envolvendo dispositivos móveis pessoais vão desde a anexação de anotações digitais a objectos físicos [105], passando por ambientes com ecrãs públicos partilhados [106], a interacções com aplicações dos mais variados tipos [107].

O uso destes dispositivos pessoais, para questionar e controlar objectos em ambientes inteligentes, torna-se atractivo devido a alguns aspectos que podem ser encontrados nos dispositivos actuais:

- **Comunicação *Wireless*:** Além da cada vez maior área que as redes cobrem, os operadores móveis oferecem cada vez mais serviços de comunicação, tais como GPRS, EDGE, ou UMTS, que podem providenciar comunicações rápidas, fiáveis e económicas. Por outro lado, protocolos de comunicação de curto alcance, como o Bluetooth, permitem que sejam estabelecidas comunicações ad-hoc locais entre dispositivos similares.
- **Detecção de *Tags*:** A recente adição da tecnologia NFC permite que os dispositivos detectem e leiam etiquetas RFID passivas (compatíveis com NFC). Uma das empresas que mais tem apostado nesta tecnologia tem sido a finlandesa Nokia, tendo até à data desenvolvido alguns telemóveis que já se encontram no mercado. Para a Nokia, é possível que esta tecnologia tenha um grande impacto no comércio móvel (*mcommerce*).  
A detecção de *tags* também pode ser efectuada através das vulgares câmaras fotográficas dos dispositivos que podem ler *visual Tags* 2D, permitindo que estas últimas "enviem" informação para o dispositivo móvel, tal como referido na subsecção 3.3.2.
- **Recursos computacionais:** Os telemóveis e os PDAs têm-se tornado em poderosos dispositivos computacionais, possuindo cada vez melhores processadores, cada vez mais memória RAM e memória Flash. Uma vez que necessitam de muita energia para as funções básicas de comunicação de longo alcance, as suas baterias suportam bem o uso de aplicações e comunicações de curto alcance. Visto que os utilizadores se encontram habituados a recarregar regularmente os dispositivos, estes apresentam uma energia virtualmente ilimitada, estando sempre ligados.
- **Ecrã e teclado programáveis:** São já muitos os dispositivos que possuem ecrãs com

dimensões agradáveis e teclas programáveis, botões de navegação, rodas de navegação, ou até *touchscreens*, permitindo que se possam criar interfaces atraentes e versáteis para as aplicações.

Ao longo dos anos têm surgido produtos comerciais que conjugam os aspectos focados, resultando em propostas muito interessantes e práticas (e.g., os telecomandos PDA da gama Pronto<sup>25</sup> da Philips). Já Nichols et al. propuseram o uso do PDA como controlador pessoal universal [108] e Zimmermann et al. desenvolveram a ideia da consola remota universal em [109].

Assim sendo, verifica-se que as características referidas levam os dispositivos móveis pessoais a oferecer novas formas de controlo para as aplicações de ambientes de Ubicomp. Por um lado, podem fornecer informação adicional sobre o estado da aplicação, ou sobre certos detalhes, caso não existam outros ecrãs, ou estender de forma mais pessoal os existentes. Por outro lado, podem estender ou personalizar a interface com o utilizador do sistema, especialmente em situações nas quais o espaço de aplicação seja limitado. Existe ainda a possibilidade de serem usados como meio de identificação dos seus utilizadores pois ao serem detectados num determinado ambiente indicam, com uma pequena margem de erro, que o seu utilizador se encontra presente.

## 3.7 Ecrãs

A cada vez maior necessidade de espaço de visualização resultante de aplicações mais exigentes em termos gráficos, e os desenvolvimentos na área da Ubicomp têm levado a uma procura de soluções de visualização alternativas.

São várias as tecnologias que têm vindo a ser investigadas, e propostas de futuros ecrãs tendem a envolver uma mudança de paradigma para longe da realidade actual, na qual os computadores têm os seus próprios ecrãs fixos. Esta mudança vai em direcção ao uso de múltiplas superfícies que já existam no ambiente. Welch et al., com [110], são um bom exemplo de quem vê as superfícies do dia-a-dia, tais como secretárias, mesas, paredes, entre outras, a tornarem-se nos ecrãs consistentes do futuro. A ubiquidade associada a este tipo de ecrãs faz deles instrumentos poderosos, mesmo que em alguns casos seja pouca a quantidade de informação que possam vir a mostrar.

Naturalmente que, para uma integração actual de ecrãs nos sistemas de Ubicomp, outras soluções são encontradas em face daquilo que neste momento se conhece. Existem diversas aproximações quanto a este tema, destacando-se algumas tecnologias que se consideram indicadas

---

<sup>25</sup>Mais informação em: <http://www.pronto.philips.com/>

face ao que tem sido a investigação na área [111]:

- **Monitores TFT/LCD:** Os monitores comuns de tecnologia TFT/LCD (*Thin Film Transistor/Liquid Crystal Display*) continuam a ser a primeira solução para muitas das aplicações de UbiComp que não necessitam de uma visualização em condições extraordinárias. Estes monitores são fáceis de encontrar e de aplicar, podendo ser adquiridos a bons preços dada a grande variedade de marcas existentes no mercado.
- **Ecrãs de rede:** Os ecrãs de rede<sup>26</sup> (ou *IP-enabled displays*) apresentam um endereço IP (*Internet Protocol*) com o qual se identificam numa rede Ethernet. Estes ecrãs permitem que informação seja distribuída por diversos pontos de um ambiente sem que para tal tenha que existir um computador associado a cada ponto. Elimina-se a necessidade de se ter um PC externo dedicado ao ecrã, apresentando-se uma integração perfeita com as redes IP de forma a facilitar a capacidade de distribuição de uma variedade de conteúdos por vários ecrãs. São já algumas as grandes empresas que apostam no desenvolvimento destas soluções de visualização, sendo exemplos a Planar<sup>27</sup>, a BenQ<sup>28</sup> e a ViewSonic com a linha de 42" LCD ND42<sup>29</sup>. Estes ecrãs possuem processador, sistema operativo próprio, e ligando-se a um servidor podem requisitar, ou ser-lhes enviado, conteúdos de acordo com a sua audiência.
- **Ecrãs de parede:** Ecrãs do tamanho de paredes podem oferecer soluções de alta resolução, sendo exemplos deste tipo de ecrãs o DataWall do MIT [112], o Scaleable Wall da Princeton [113], o ecrã Wall da NCSA [114], ou o LionEyes da PennState [115]. Preferencialmente, são soluções indicadas para aplicações nas quais não devem passar despercebidos, e para uso em interacções activas entre utilizadores em ambientes privados.
- **VRP (*Virtual Rear Projection*):** Os VRP podem ser uma solução para ambientes dinâmicos, onde diferentes configurações de mobiliário podem facilmente obscurecer grandes zonas de visualização. Algumas soluções deste tipo podem ser encontradas em [116] e em [117], nas quais são usados múltiplos projectores a partir de diferentes posições, com projecções sobrepostas para criar a chamada "*Virtual Rear Projection*".
- **Sistema Projector-Câmara Orientável:** Este tipo de sistema é uma outra solução para

<sup>26</sup>Do anglo-saxónico *network displays*.

<sup>27</sup>[http://www.planar.com/products/network\\_displays/nd/index.html](http://www.planar.com/products/network_displays/nd/index.html)

<sup>28</sup><http://www.benq.com/products/LCDTV/>

<sup>29</sup><http://www.viewsonic.com/products/largescreendisplays/networkdisplays/>

o problema da oclusão em projecções frontais. A projecção é feita para uma multitude de superfícies a partir de um único projector fixo. O sistema Everywhere Displays (EDs) [118], que pode ser colocado no tecto de um espaço, utiliza um projector LCD/DLP e um espelho *pan/tilt* controlado por computador para permitir a projecção do ecrã em qualquer plano de superfície dentro da área de projecção do sistema.

- **Projectores móveis:** Este sistema é uma extensão ao EDs, funcionando como projector-câmara do tipo *handheld*. É uma versão móvel colocada dentro de uma *toolbox* [119]. De todos os modos, o sistema ainda está dependente da ligação de um cabo para dar energia ao projector.
- **E-Paper:** Ao contrário de um ecrã convencional plano, que usa luz traseira para iluminar os seus *pixels*, uma versão do *e-paper* reflecte a luz como o papel, tendo a capacidade de manter o texto e imagens indefinidamente sem ter de utilizar electricidade para o efeito. Estes ecrãs apresentam como características interessantes poderem ser dobrados, como o papel tradicional, e embutidos nas superfícies frontais de objectos.

## 3.8 Projectos e Sistemas Relacionados

Nesta secção, são apresentados alguns trabalhos de implementação na área da Ubicomp, que partilham com o tema desta dissertação objectivos e ideias essenciais, providenciando uma base interessante para o trabalho desenvolvido. Deste modo, quer seja pela arquitectura proposta, pelas aplicações desenvolvidas, pelas tecnologias utilizadas, ou até pela exploração das ideias de interacção, os trabalhos abordados são um contributo importante para a investigação desenvolvida.



Figura 3.9: (esquerda) BlueBoard e (direita) GroupCast: Em contexto de utilização.

Um primeiro trabalho a considerar é o **BlueBoard** [120] (ver Figura 3.9), uma arquitectura desenhada para suportar encontros rápidos, "leves", com os seus utilizadores de modo a providenciar um acesso aos seus conteúdos pessoais. É suportado por um grande ecrã público e funciona com a passagem de um *badge* através de um leitor dedicado para que se efectue a autenticação do utilizador. O ecrã apresenta ao utilizador conteúdos encadeados, como calendários pessoais, que são navegados através de uma interface do tipo *touch-screen*. É de realçar que os conteúdos não são alterados de forma dinâmica.

Por outro lado, o projecto **GroupCast** [121] (ver Figura 3.9) utiliza *badges* de infra-vermelhos para detectar audiências junto de um ecrã público. Assim, pode apresentar conteúdos no ecrã de modo a iniciar ou a melhorar conversações entre indivíduos ou grupos que se situam junto ao mesmo. O conteúdo correcto é automaticamente seleccionado de acordo com perfis e preferências existentes, e definidos para cada utilizador na área (ambiente) onde o sistema é implementado.

No caso do **Hermes Photo Display** [122] (ver Figura 3.10), este é desenhado com o objectivo de permitir que os seus utilizadores partilhem fotografias uns com os outros. Para tal, necessitam de ter um telemóvel com *Bluetooth* através do qual fazem o *upload* e o *download* das fotografias para e do ecrã público. Para este projecto, o facto de muitos telemóveis terem a capacidade de partilhar dados multimédia via Bluetooth, não necessitando para tal de um *software* específico, é uma vantagem a ser aproveitada.

Uma vez mais, tal como no caso do BlueBoard, o conteúdo apresentado não é dinâmico, requerendo uma interacção directa por parte do utilizador para fazer a navegação pelas fotografias.

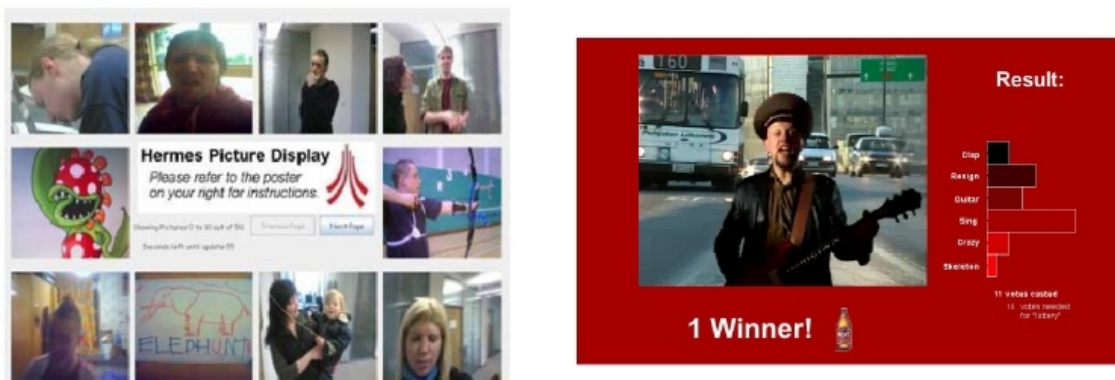


Figura 3.10: (esquerda) Hermes Photo Display e (direita) Mobilenin: Apresentação de informação no ecrã.

Tal como o anterior, o **MobiLenin** [123] (ver Figura 3.10) é um sistema que também utiliza

### 3.8. PROJECTOS E SISTEMAS RELACIONADOS

---

telemóveis para a interacção com um ecrã público, mas via HTTP/GPRS. O sistema suporta a edição colaborativa de um vídeo musical que é composto por seis faixas lineares com exactamente o mesmo comprimento mas com diferentes conteúdos. Num determinado momento apenas uma delas é visível. Os utilizadores podem votar numa das faixas e aquela que obtiver mais votos será mostrada.

Este sistema utiliza *software* específico no lado dos clientes, isto é, nos telemóveis dos utilizadores.

O **BluScreen** [124] (ver Figura 3.11) é um outro projecto usando como aproximação o uso de dispositivos móveis (telemóveis, PDA's) para influenciar os conteúdos de um ecrã público. Utiliza dispositivos com Bluetooth para detectar a audiência.

Os utilizadores são levados a interagir de uma forma passiva, uma vez que eles influenciam na amostragem dos conteúdos apenas por serem detectados na área de implementação do sistema. O BluScreen procura maximizar a exposição à audiência corrente em oposição à apresentação de conteúdos que vão de encontro a preferências conhecidas de utilizadores. Se assim não fosse, seria necessário que o sistema tivesse um conhecimento prévio da audiência, o que não é um dos seus objectivos.

A actividade de co-presença dos dispositivos é utilizada para leiloar tempo de visualização num ecrã público para agentes de propaganda autónomos. A interacção passiva do utilizador com o ecrã, tal como, parar na zona para ver os conteúdos, é cooptada para acidentalmente providenciar *feedback* para o "mercado de negócios" dos agentes sobre preferências implícitas, alterando assim os conteúdos que um utilizador possa receber no futuro.

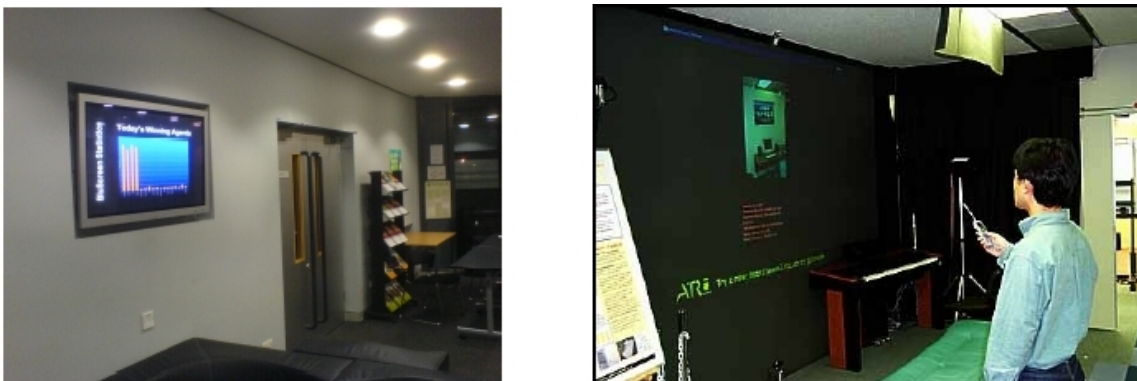


Figura 3.11: (esquerda) BluScreen e (direita) iRemote: Em contexto de utilização.

A implementação de uma aplicação colaborativa é o objectivo do projecto **iRemote** [125] (ver Figura 3.11). Este tipo de aplicação tem em consideração aspectos relacionados com en-

tretenimento e com *communityware*. Um exemplo de cenário deste tipo é permitir que pessoas façam *graffiti* digital em ecrãs públicos, usando comandos nos respectivos telemóveis.

Para funcionar, este sistema não necessita que os telemóveis tenham previamente instalado *software* específico, tendo os utilizadores que aceder a um endereço URL associado com o ecrã. Existe uma interacção directa e activa com o sistema via HTTP/GPRS. Os conteúdos visualizados no ecrã são dinamicamente actualizados através de hiperligações e formulários que são seguidos pelos utilizadores quando acedem às páginas HTML.

O **magicmirror** (ver Figura 3.12), desenvolvido pela Thebigspace<sup>30</sup>, é um sistema comercial já em funcionamento em lojas Prada, em Milão. Este sistema tem por base um espelho que também é um ecrã táctil, com computador, leitor de *tags* RFID, e retransmissor de comunicações Wi-Fi, tendo por função elementar dar informação adicional sobre as peças de roupa experimentadas pelos clientes. Deste modo, este sistema aumenta objectos, as peças de roupa, através da disponibilização de informação num ecrã e utilizando tecnologia RFID para identificação automática e não intrusiva.

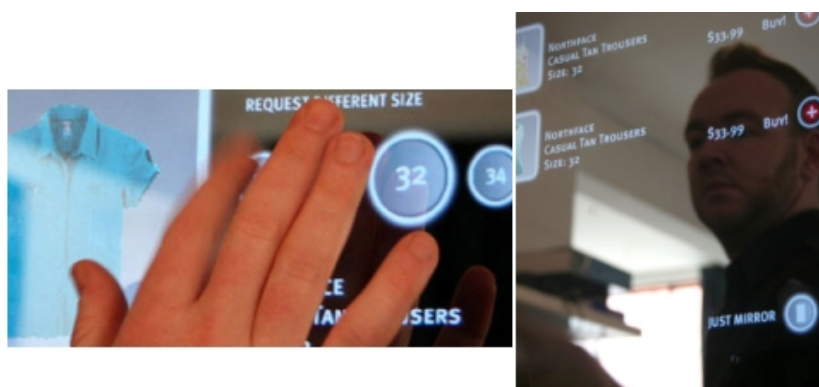


Figura 3.12: magicmirror: Funcionalidades disponibilizadas ao utilizador.

A empresa portuguesa de produção têxtil Throttleman visa dotar as suas lojas com sistemas magicmirror [11]. Para tal, colocará o sistema nos provadores de lojas para identificar automaticamente as peças levadas por cada cliente, que a partir daí poderá aceder a várias funcionalidades disponibilizadas através do ecrã táctil do espelho. Uma das funcionalidades será a realização de pedidos que seguem via Wi-Fi directamente para um dispositivo de um funcionário.

Em Itália, a Universidade romana La Sapienza, através do seu centro RFID Lab, desenvolveu uma solução também baseada em tecnologia RFID para o Museu das origens do Homem<sup>31</sup> (*Museo delle Origini*). Foi criado um ambiente inteligente que aumenta as peças em exibição,

<sup>30</sup><http://www.thebigspace.com/>

<sup>31</sup><http://www.museoorigini.it/>

ao qual foi dado o nome **Wi-Art** [126]. O sistema permite que os visitantes recebam num PDA, que podem utilizar ao longo da visita, informação detalhada sobre as peças quando se aproximam das mesmas. Dados relevantes são armazenados nas *tags* RFID colocadas nas peças, sendo automaticamente enviados para os PDAs, equipados com leitor RFID, sempre que estes se aproximam das peças, e originando que informação adicional sobre as mesmas seja mostrada no ecrã dos PDAs.

Além do uso de tecnologia RFID e de dispositivos pessoais móveis como os PDAs, é ainda de realçar que o sistema utiliza tecnologia Intel e tecnologia Oracle para o sistema de gestão da bases de dados, com especial referência para o pacote *middleware* "Oracle Sensor Edge Server", que se encarrega da gestão dos leitores RFID.

A informação sobre as peças varia entre a descrição textual e os conteúdos vídeo e áudio, utilizando também tecnologia "text-to-speech" para que os conteúdos textuais sejam falados para visitantes invisuais. Assim sendo, o sistema permite que dados adicionais, previamente compilados, sobre as peças sejam transformados em informação, aumentando a qualidade e a capacidade educacional de cada uma das visitas.

**DIABlu** (*Digital Arts' Bluetooth*) [127] é outro projecto nacional que tem como objectivo o desenvolvimento de uma infra-estrutura computacional para detecção de pessoas num ambiente. Em função dessa detecção podem ser criadas as mais variadas respostas artísticas enquanto sistemas de aplicação que usam a infra-estrutura. O sistema foi desenvolvido pela Escola das Artes da Universidade Católica do Porto, visando uma integração fácil em aplicações de instalações de arte digital.

O projecto DIABlu tem por base a utilização da tecnologia Bluetooth e os dispositivos móveis pessoais que implementem a referida tecnologia. Deste modo, o sistema faz a detecção dos utilizadores através dos seus dispositivos com Bluetooth. É também essencial para o projecto a utilização do protocolo OSC (*Open Sound Control*), pois muitas das aplicações artísticas que podem utilizar a infra-estrutura utilizam-no, e a integração com tecnologias como Flash, Processing e Pure Data, usadas para o desenvolvimento das aplicações. A infra-estrutura é basicamente dividida em 3 módulos: (1) o servidor, que faz a gestão de detecções; (2) o cliente, que é o dispositivo móvel com Bluetooth; (3) e a aplicação final que recebe mensagens do servidor através do protocolo OSC. Além do cenário base, em que o servidor apenas detecta os utilizadores, existe ainda um cenário alternativo que permite a interacção entre o servidor e os dispositivos móveis, desde que suportem Java e implementem a API JSR-82 uma vez que a aplicação móvel é desenvolvida em J2ME e utiliza Bluetooth para enviar comandos para o

servidor.

A infra-estrutura DIABlu já foi experimentada, por exemplo, no projecto "Nulltidão", uma instalação multimédia que integra um painel, subdividido em seis ecrãs, e uma *webcam* colocada no topo do mesmo para captar as imagens. O DIABlu detectava o número de dispositivos presentes no ambiente, que influenciavam a definição da imagem visualizada nos ecrãs, na medida em que quantos mais presentes, mais difícil seria entender a composição formada pelos vários ecrãs.

Em [128] pode ser encontrado um projecto, **eCampus** (ver Figura 3.13), de desenvolvimento de uma infra-estrutura pervasiva a ser aplicada no campus da Universidade de Lancaster. A infra-estrutura envolve vários tipos de ecrãs (plasma, LCD e sistemas de projecção), utiliza Wi-Fi e Bluetooth para as comunicações e ainda sensores (as câmaras dos telemóveis, entre outros), encontrando-se aplicada em alguns espaços públicos ao longo do campus. Além de ser usado em eventos artísticos, o sistema oferece notícias locais e outros serviços de informação. Como funcionalidade mais interessante, para a presente tese, o sistema apresenta uma possibilidade de interacção entre utilizadores, com dispositivos móveis pessoais com câmara integrada, e os ecrãs.

O sistema utiliza um módulo designado por **iCapture** [129] (ver Figura 3.13, imagem da direita) que é uma aplicação que se instala nos dispositivos dos utilizadores, tendo por objectivo capturar, através da câmara, *visual tags* disponibilizadas nos grandes ecrãs e que correspondem a artigos que podem ser consultados (ver Figura 3.13). A aplicação faz o reconhecimento da *tag* e produz um objecto com informação que é passado para um servidor Web via HTTP/GPRS. Neste reside uma outra aplicação que, além de controlar a interface com os ecrãs, processa o pedido e questiona um segundo servidor responsável pela produção de RSS e agregação dos *feeds* RSS. De acordo com a informação que lhe é passada, este servidor responderá enviando a hiperligação do respectivo artigo para ser visualizado em formato HTML no dispositivo pessoal, respondendo assim ao pedido inicial desencadeado pelo último.

O projecto eCampus optou por *visual tags* e pela câmara como sensor para fazer a interacção entre os ecrãs e utilizadores, fazendo ainda uma distinção entre informação que é visualizada nos ecrãs públicos partilhados e aquela que pode ser visualizada nos ecrãs mais pequenos dos dispositivos pessoais móveis.

Um outro trabalho também relacionado com a presente tese é o proposto por Bardram et al. em [130], onde apresentam um sistema "*context-aware*" aplicado a um ambiente hospitalar, mais concretamente a uma ala de cirurgias. O sistema **AWARE** (ver Figura 3.14) é composto por: um módulo de rastreamento de localizações, tanto de clínicos como de pacientes;

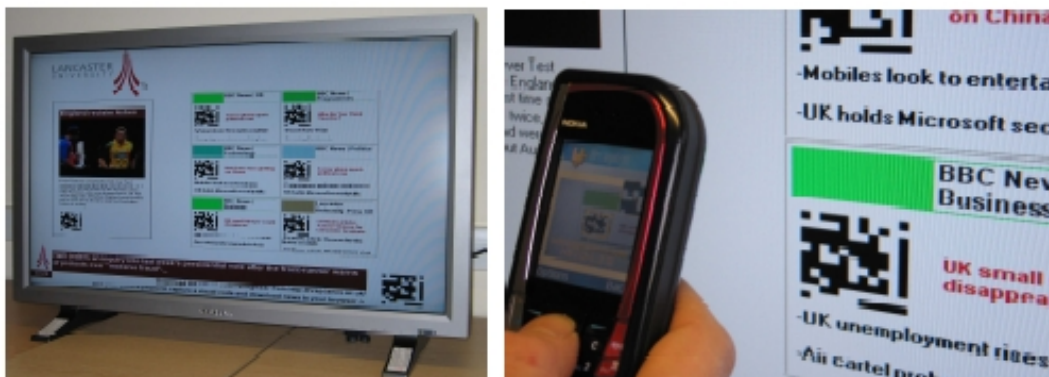


Figura 3.13: eCampus: (esquerda) Um ecrã grande e (direita) aplicação iCapture num dispositivo móvel.

uma infra-estrutura de aquisição, gestão e distribuição de informação contextual; uma aplicação (AwareMedia) de agendamento de operações "context-aware" que funciona num grande ecrã público; e uma aplicação (AwarePhone) para telemóveis com o intuito de informar o utilizador sobre actividades de colegas e salas de operações.



Figura 3.14: Sistema AWARE: Em contexto de utilização.

A arquitectura central que gere a informação contextual do sistema utilizou a JCAF<sup>32</sup> (*Java Context-Awareness Framework*) na modelação das entidades e da informação contextual. Sendo o coração do sistema, entre várias funções, este módulo é responsável por receber os dados provenientes dos rastreamentos e também pelo envio de informação para os telemóveis e para os ecrãs públicos. O sistema de rastreamento baseia-se em Bluetooth. Assim, em determinados pontos colocaram PC's estacionários, possuindo cada um deles um adaptador USB para Bluetooth. Quem não possuir um telemóvel com tecnologia Bluetooth usa uma *tag* Bluetooth ao pescoço. A aplicação AwareMedia corre num grande ecrã, permitindo que ocorram interacções

<sup>32</sup><http://www.daimi.au.dk/~bardram/jcaf/>

entre os clínicos e o sistema, e fornecendo acesso às mais variadas informações contextuais, como por exemplo, visualizar onde se encontra (mais concretamente, qual foi o último ponto por onde passou) cada utilizador e aceder ao agendamento das actividades globais. Os telemóveis dos clínicos, além de serem utilizados como *tags* para detecção da sua localização, possuem a aplicação AwarePhone em execução, permitindo-lhes acederem a algumas informações contextuais importantes, como por exemplo, onde se encontram determinados colegas.

As *tags* Bluetooth são disponibilizadas à entrada do ambiente, podendo ser adjudicadas diariamente aos utilizadores através da aplicação AwareMedia. No início do projecto foram distribuídos, aos clínicos, 15 telemóveis Nokia com sistema Symbian, um requisito para a aplicação AwarePhone, e com tecnologia Bluetooth.

Um outro projecto que utiliza tecnologia RFID é o *Augmented Knight's Castle* [131] (ver Figura 3.15) que visa enriquecer, de forma computacionalmente pervasiva, um *playset* (conjunto de jogo) da marca Playmobil. O conjunto "*Knight's Empire Castle*" serviu como aplicação da infra-estrutura proposta pelo projecto, mas, para os autores, qualquer um outro cenário poderia ser opção para a aplicação. Neste exemplo aplicacional, o cenário de jogo é educacionalmente enriquecido e "aumentado" através de música de fundo, efeitos de som e comentários verbais de peças que reagem ao jogo praticado pelas crianças.

A tecnologia RFID é mais uma vez utilizada para a identificação automática e não intrusiva, uma vez que se pretende não interferir com ou comprometer o mecanismo tradicional de jogo. Deste modo, são colocadas *tags* RFID nas peças que compõem o conjunto e que foram escolhidas para actores do cenário "aumentado" (ver Figura 3.15). É interessante referir que o número de *tags* colocadas nas peças varia até um máximo de 3 (na cabeça, nas costas, e nos pés) para que a detecção seja o mais apurada possível.

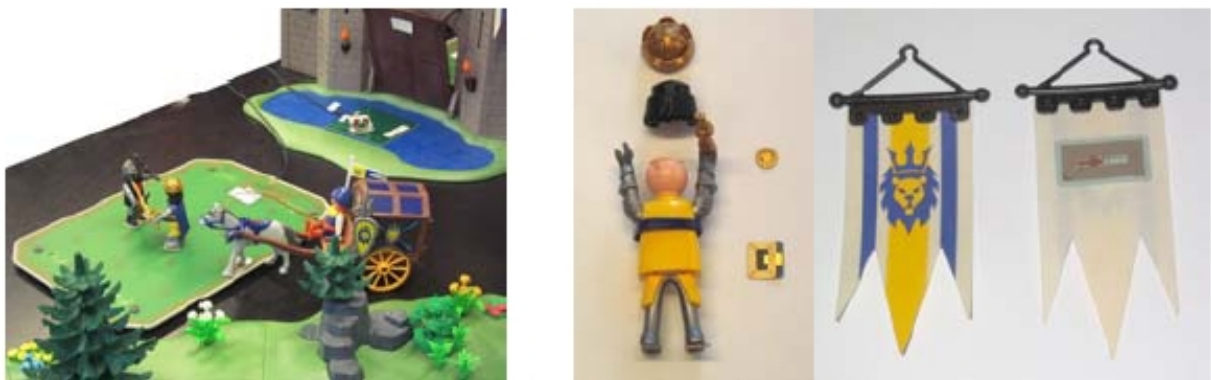


Figura 3.15: Knight's Castle: (esquerda) Antenas RFID embutidas no *playset* e (direita) *tags* RFID colocadas em peças.

Há ainda dispositivos móveis que são equipados com leitores RFID, sendo utilizados para provocarem interações e para melhorarem o jogo. Foram utilizados módulos de mini leitores RFID controlados por BTnode (Skyetek M1). Os módulos são anexados a telemóveis Nokia 6830, comunicando através de Bluetooth, e ainda a algumas peças, que têm também a adição de uma placa com sensores de luz e aceleração 3D e com microfone.

Os dados RFID de todos os leitores presentes no cenário são filtrados, agregados e enriquecidos, com informação adicional sobre as peças, através de um módulo *middleware*, sendo guardados enquanto objectos num modelo de localização simbólico. Assim sendo, consegue-se obter em cada momento o estado global de localizações, podendo-se aumentar as peças, e consequentemente o jogo, com informação adicional, não mostrada num ecrã, mas sim transmitida através de áudio.

Já o projecto **Cooltown** [22], desenvolvido nos HP Labs, integra-se nas áreas de construção e composição de serviços, além da validação das experiências dos utilizadores. O Cooltown visa atribuir páginas Web e/ou serviços Web a todas as pessoas e a todos os objectos de forma a que serviços computacionais estejam por todo o lado. Com este projecto, foram sistematicamente integrados *web services* para acentuar as comunicações com pessoas móveis, providenciando serviços específicos da localização nos locais visitados, além de interação com os objectos encontrados. De modo a ajudar tanto utilizadores como programadores a formular um modelo comum sobre como estes sistemas se comportam, estes são chamados de "presença na Web". Presença na Web corresponde à representação de pessoas, locais e objectos na Web. Deste modo, o conceito de "*home page*" é alargado para incluir todas as entidades físicas e uma correlação deliberada e automática, suportada no sistema, da *home page* (a presença na Web) com a entidade física.

O objectivo do projecto é providenciar uma infra-estrutura para computação nómada (um termo usado no projecto para computações ubíqua e móvel orientadas para pessoas). Para acederem aos serviços Web disponibilizados pelo sistema Cooltown, assume-se que as pessoas usam dispositivos com conectividade *wireless* e equipados com sensores, tais como telemóveis ou PDAs.

Um dos primeiros projectos relacionado com visitas a museus foi o **Rememberer** [132] (ver Figura 3.16), que é parte integrante do projecto Cooltown da HP. Rememberer funciona como uma ferramenta para a captura de experiências pessoais enquanto se visita um museu. O projecto tem a parceria do museu Exploratorium<sup>33</sup>, em San Francisco, onde foi conduzido um estudo

---

<sup>33</sup><http://www.exploratorium.edu/>

experimental. Rememberer permite a construção de um registo da visita, podendo ser consultado durante ou após a mesma. Adoptando o conceito do Cooltown, o registo criado pelo Rememberer consiste em páginas Web acerca das exposições visitadas, incluindo anotações e fotografias tiradas ao longo da visita. Pelos resultados obtidos, com a experiência de implementação no museu, o projecto atingiu um requisito fundamental, na medida em que o sistema não colidiu com os aspectos considerados naturais numa visita a um museu.

O sistema Rememberer consiste em: um módulo chamado "remember-this", que é composto por tecnologia com a qual o utilizador selecciona as exposições e/ou obras que pretende registar; o módulo registo da visita, que consiste no conjunto de páginas Web com informação sobre o que foi seleccionado; e um artefacto físico que relembra o utilizador acerca da visita e que contém um apontador (URL) para o respectivo registo. Para que o registo fosse mais específico em termos pessoais, foram instaladas câmaras junto das exposições para que quando um utilizador o deseje, fazendo gestos, estas tirem uma sequência de fotografias captando assim o momento em que o utilizador visita a exposição.

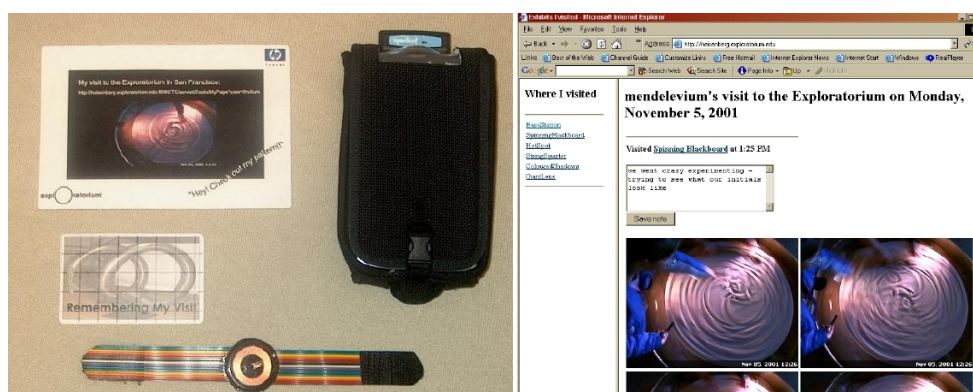


Figura 3.16: Rememberer: (esquerda) Equipamento e (direita) exemplo de uma página Web gerada.

Os testes iniciais foram realizados utilizando tecnologia RFID, com os utilizadores a possuírem etiquetas passivas RFID, na forma de cartões ou incorporadas em relógios, que ao passarem a 10 cm de um leitor colocado numa exposição indicariam que essa visita seria para registar. Como alternativa, também foram usados PDAs da HP, com comunicação Wi-Fi, colocados numa caixa, para protecção e para ocultação do ecrã, que ao serem apontados para um *beacon* de IR montado na exposição iniciava o processo de registo. Pelos resultados obtidos, o uso da tecnologia RFID tornava o sistema menos intrusivo, mais "leve" para o utilizador, mas com custos mais elevados na implementação pois junto a cada leitor, em cada exposição, estava um PC dedicado.

Em [133] especifica-se a infra-estrutura **ActivitySpot** (ver Figura 3.17), que visa suportar actividades localizadas desempenhadas por visitantes ocasionais dos espaços de aplicação. Os autores do projecto, Pinto et al., têm o objectivo de possibilitar uma experiência de utilização centrada na actividade e que use ao máximo as possibilidades de recursos disponíveis para Computação Ubíqua, não pretendendo impor qualquer requisito tecnológico específico. Nas implementações realizadas, em eventos científicos ou culturais, as actividades tiveram em consideração, por exemplo, as acções de orientar, supervisionar estudantes, jogar um jogo, partilhar conteúdos, visualizar programa de um evento ou votar num evento público.

Os visitantes (utilizadores) são levados a interagir com o sistema implementado através do seu próprio telemóvel ou PDA, de ecrãs públicos situados, e de outras tecnologias acessórias, tais como, RFID, barras de códigos, *visual codes*, ou câmaras. Enquanto que, por exemplo, as *tags* RFID servem para estimular o sistema, outras tecnologias (Web/WAP, ecrãs públicos ou SMS) têm por função tornar a experiência de utilização mais enriquecedora para o visitante.



Figura 3.17: ActivitySpot: Em contexto de utilização.

É de salientar que nesta infra-estrutura os utilizadores são levados a terem um papel activo, interagindo com o sistema implementado por via das suas acções. Sendo assim, um ponto importante a reter na implementação é o registo prévio de um perfil pessoal para cada utilizador. Este registo pode ser feito numa zona de recepção, onde os seus dados são armazenados numa base de dados que também os relaciona com as suas acções/actividades. O perfil contém o número do dispositivo pessoal a utilizar, a identificação de *tags* RFID (distribuídas para interacção) ou o endereço Bluetooth do dispositivo. Esta identificação será usada pelo sistema para que quando sentir um estímulo, uma acção desencadeada, conseguir relacionar a mesma com o seu autor. Dependendo da acção, a resposta do sistema poderá ser personalizada, como por exemplo, mostrar informação num ecrã direccionada para o perfil do utilizador.

## Capítulo 4

# Princípios e Arquitectura

Neste capítulo, identificam-se e descrevem-se princípios que servem de alicerces à infra-estrutura desejada e apresenta-se uma proposta de arquitectura, seguida nos protótipos desenvolvidos.

### 4.1 Princípios

Para uma apresentação adequada da infra-estrutura desejada, é necessário que se estabeleçam princípios considerados elementares para o desenho da mesma. Assim sendo, pretende-se que a infra-estrutura assente na co-presença dos utilizadores no ambiente dos objectos a aumentar, originando interacções acidentais. Num primeiro cenário elementar pretende-se assegurar que apenas com a detecção da presença do utilizador na área pertencente ao objecto, este possa ser aumentado com informação contextual visualizada num ecrã público. Existirá uma relação de um para um entre um objecto, ou parte dele, e um ecrã. Um objecto pode ser uma coisa física, inanimada e facilmente visualizada, ou até pode ser uma instituição, ou mesmo um outro utilizador com actividade no ambiente. Apesar de, numa fase inicial, não ter existido a preocupação em implementar sistemas de informação ambiental, verifica-se que a infra-estrutura se deve dirigir para esta área de investigação pelas características que apresenta. Por último, a infra-estrutura deverá ter em consideração um segundo nível de interacção. Apesar de desencadeada pela interacção ainda "acidental" vinda da co-presença do utilizador no ambiente, poderá existir uma interacção activa por parte deste com o par objecto/ecrã, utilizando-se para esse fim os vulgares dispositivos móveis pessoais, como telemóveis e PDAs. Estes dispositivos permitem ainda que a informação possa ser visualizada de um modo privado. A Figura 4.1 mostra como estes princípios são considerados num cenário genérico.

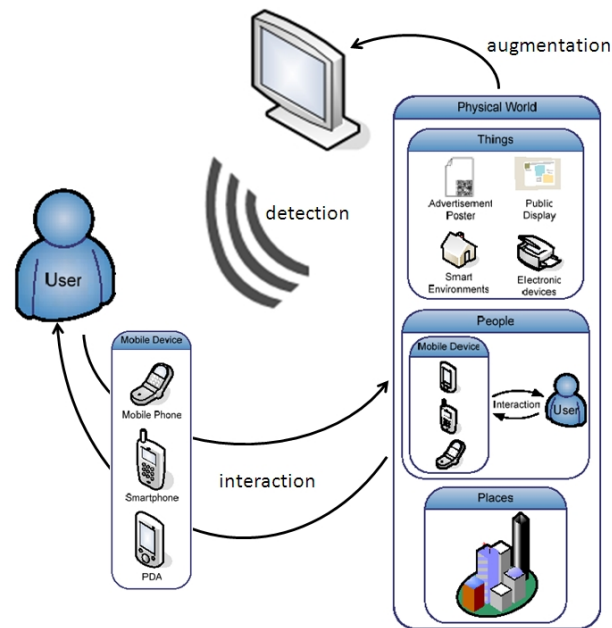


Figura 4.1: Cenário Genérico: Detecção (do utilizador), aumentação visual (do objecto) e interacção activa (entre utilizador e objecto) [adaptado de [10]].

Deste modo, a noção de sistema de informação ambiental, as interações acidentais num ambiente, o uso de ecrãs públicos, a informação contextual e os dispositivos móveis servem de linhas de orientação para o tipo de proposta que se pretende apresentar com este trabalho.

#### 4.1.1 Sistema de Informação Ambiental

A ideia de apresentar informação de uma forma tranquila e naturalmente integrada no ambiente tornou-se apelativa desde a formulação da Ubicomp, sendo um princípio a seguir por esta tese. Considerada um sub-domínio da UbiComp, tem como premissa base a apresentação de informação através de alterações suaves e integradas no ambiente, permitindo que os utilizadores não sejam distraídos das suas tarefas centrais, mas que possam receber e estar cientes de informação adicional e não crítica. São vários os termos que caem neste sub-domínio, como por exemplo *"peripheral display"*[134], *"notification system"*[135] ou *"ambient information system"*[136]. Sendo esta uma área de investigação recente, não existe ainda uma definição aceite de forma universal para estes sistemas. Desde Ishii et al. [44] a Matthews et al. [134], passando por Mankoff et al. [137] e Pousman et al. [136], todos eles apresentam propostas de definição com vários pontos em comum. São os últimos que tentam unificar os vários termos propondo o nome de Sistema de Informação Ambiental<sup>1</sup> e definindo as seguintes características para estes

<sup>1</sup>Do anglo-saxónico *"ambient information system"*.

sistemas:

- Visualização de informação considerada importante mas não crítica.
- Movimentação da atenção da periferia para o centro, e vice-versa.
- Integração esteticamente agradável e ambientalmente apropriada.
- Apresentação de mudanças subtis para reflectir actualizações na informação (não deve ser foco de distracção).
- Utilização de meios já existentes no ambiente para apresentação da informação.

Uma vez que estes sistemas têm vindo a ser desenvolvidos para diferentes áreas de aplicação, cada uma com os seus requisitos específicos, são vários os tipos de desenho adoptados para apresentação da informação, variando desde os ecrãs aos produtos tangíveis, com os últimos a serem a solução mais desejada.

#### 4.1.2 Interações Acidentais e Co-Presença

Existem ocasiões em que o utilizador não consegue ou nem é requerido que centre a atenção no dispositivo computacional. Estas são as situações que mais interessam para esta tese.

Segundo a literatura tradicional relacionada com a área da interacção pessoa-máquina, o utilizador tem um objectivo em mente quando decide interagir com um computador. Isto é frequentemente caracterizado pelo modelo de interacção de Norman [138], o qual consiste em duas fases primárias: execução e avaliação. Em contraste com este modelo, a noção de "interacções acidentais" foi proposta por Dix para descrever as acções que são co-optadas por um sistema para servirem uma finalidade diferente daquela que o utilizador tem em mente [139]. Considera-se a interacção acidental, como a situação onde acções executadas com um outro propósito (ou sinais inconscientes) são interpretadas de modo a melhorar futuras interacções no quotidiano humano [140].

Esta noção é importante para o presente trabalho, pois pretende-se que a infra-estrutura parta da interacção normal de um utilizador com um objecto para que este seja aumentado, automaticamente e sem esforço adicional claro por parte do utilizador, com a informação digital adicional. Saliente-se que acidentalmente não implica que uma acção, ou subsequente objectivo, não seja intencional, podendo o utilizador ter especificado previamente ao sistema a sua intenção e não pensar conscientemente na mesma enquanto executa as acções. Interações acidentais também diferem em termos do *feedback* recebido pelos utilizadores. Numa aplicação tradicional, o *feedback* recebido pelo utilizador acerca das suas acções será directo, explícito. No entanto,

numa interacção dita accidental, esse *feedback* poderá ser, em muitos casos, mínimo, discreto e até retardado. Estas características são importantes para que os resultados de um sistema deste tipo não interfiram com a actividade central do utilizador que se desenrola no ambiente do sistema.

Já o termo co-presença refere-se às condições espaço-temporais nas quais as pessoas interagem umas com as outras. Resumindo, a co-presença pode ser definida como a condição em que as pessoas têm a percepção que estão próximas o suficiente para serem percebidas naquilo que estão a fazer, incluindo a experienciação que têm dos outros [141]. Existe co-presença quando duas ou mais pessoas se encontram no mesmo local. Regra geral, não acontece de forma intencional, podendo também ser vista como uma forma de interacção accidental.

Contudo, se para os humanos a co-presença pode ser naturalmente percebida, já para um sistema computacional existirá a dificuldade na capacidade de percepção, tendo que existir um meio tecnológico que detecte a co-presença. Duas boas soluções, na perspectiva desta dissertação, passam pela classe de tecnologias de PAN, como Bluetooth, e pela identificação através de RFID. No caso das PAN, o alcance efectivo destas corresponderá a uma distância à qual em termos humanos é normal ter-se a percepção das características individuais dos outros. Já um sistema RFID pode ser projectado para que a detecção seja feita à distância que se desejar. Uma tecnologia como a Bluetooth tem ainda como vantagem o facto de ser largamente adoptada pelos vários dispositivos móveis utilizados diariamente pelas pessoas.

A utilização de uma simples etiqueta RFID ou de um dispositivo pessoal com Bluetooth por parte de um utilizador permitirá que a computação da co-presença seja "incorporada" (*embodiment*) no quotidiano. Esta noção de *embodiment* significa que através de meios de interacções físicas, o utilizador pode participar em sistemas de co-presença, sem que para tal seja necessário participar em actividades que não pertençam à sua rotina diária. Em [142], Dourish, argumenta que *embodiment* não significa apenas "manifestação física", mas estar profundamente envolvido nas experiências de todos os dias. O *embodiment* no mundo real, facilitado por uma modalidade de co-presença, vai de encontro às características de uma interacção accidental, que deverá resultar de um outro objectivo do dia-a-dia.

### 4.1.2.1 Noção do Espaço (Ambiente)

A noção do espaço será determinante para o estabelecimento do estilo de interacção que pode ser encontrada. Ajudará também a determinar o conteúdo e forma de apresentação da informação. Deste modo, para o presente trabalho, são considerados os diversos tipos de espaço

encontrados em [129]. Esses espaços têm as seguintes designações:

- Público - A noção mais lata e na qual todas as outras se podem encaixar ou basear. O espaço será uma área aberta a qualquer pessoa não apresentando quaisquer restrições.
- Privado - Uma área à qual se aplicam restrições, onde não será possível a entrada de qualquer pessoa. Um escritório ou um laboratório de investigação são apenas dois exemplos de ambientes onde existe a co-presença de pessoas conhecidas que desempenham funções interligadas. Também pode ser referido como espaço semi-público pois será público para um pequeno, co-localizado grupo de pessoas [143].
- Transiente - Espaços onde as pessoas não se congregam facilmente ou frequentemente, como são os casos de corredores ou entradas de edifícios. Dependendo do contexto, o acesso aos mesmos pode ser público ou privado.
- Social - São geralmente espaços públicos que apresentam como característica principal o propósito de desencadear interações sociais. Caso apresentem restrições, como o pagamento de uma entrada, tornam-se privados.
- Informativo - Estes espaços encontram-se intimamente ligados aos espaços transientes e são, muitos deles, espaços públicos. São espaços dedicados ou, pela quantidade de pessoas que neles transitam, são usados para anúncios e colocação de mensagens informativas.

Um espaço físico não estará obrigatoriamente contido em apenas um dos tipos apresentados por Mitchell et al. em [129]. Aliás, podem ser encontrados vários espaços que se caracterizem por serem um misto de vários tipos. Obviamente que um espaço definido como sendo público não será privado, mas qualquer um destes dois pode ser conjugado com um ou mais dos outros três.

#### 4.1.3 Ecrãs Situados

Verifica-se que os ecrãs, com as mais diversas formas e baseados em vários tipos de tecnologias (ver Subsecção 3.7), estão a tornar-se incrementalmente ubíquos, mantendo-se como a forma mais efectiva de se criar uma presença de informação digital associada ao mundo físico. Caso sejam conjugados com câmaras e/ou sensores, podem detectar o meio envolvente e deste modo fazer um cruzamento entre dados do mundo físico e dados do mundo virtual, providenciando informação contextual de maior utilidade.

Podem-se encontrar duas tendências divergentes no que toca a cenários de sistemas de computação com utilização de ecrãs situados e partilhados [18]. Por um lado, existe a corrente

ligada a ecrãs públicos situados, de preferência grandes com elevadas resoluções, como é o caso dos *digital bulletin boards* e dos *kiosks*. Existem em várias formas e escalas, incluindo ecrãs de rua (usados preferencialmente em publicidade), ecrãs de projecção (projectores usados em conferências e instituições de ensino), e pequenos ecrãs colocados junto a objectos. O uso destes ecrãs é normalmente distribuída no tempo, tendo como via de *input* (entrada para interacção) uma única modalidade que é, regra geral, fisicamente ligada ao ecrã (e.g., *touchscreens*) [18].

Por outro lado, a segunda corrente está relacionada com os dispositivos móveis, tais como telemóveis, portáteis e PDAs. Estes funcionam não apenas como entrada para interacções com os sistemas, mas também como ecrãs.

Os dispositivos móveis dispersam o controlo e o acesso pelos diversos utilizadores, apresentando um poder computacional limitado e pequenos ecrãs que prejudicam uma apresentação mais dinâmica. Já os ecrãs situados partilhados oferecem um maior poder computacional e uma maior resolução para apresentação e conjugação da informação.

São os ecrãs situados que fazem parte de um dos módulos elementares da arquitectura proposta. Numa primeira aproximação, pretende-se que a informação que aumenta um objecto seja apresentada junto ao mesmo, funcionando o ecrã como extensão directa do objecto. Deste modo, a acção de aumentar o objecto torna-se, numa primeira fase, contida, estando dependente da aproximação do utilizador, sendo menos intrusiva e não obrigando o utilizador a ter uma interacção activa.

De acordo com a sua disponibilidade e modalidade de acesso, os ecrãs na infra-estrutura podem ser categorizados da seguinte forma [144]:

- Público: O ecrã encontra-se num espaço público, sendo visualizado por todos os que acedem ao espaço e podendo ser usado por utilizadores registados no sistema.
- Partilhado: O ecrã pode ser visualizado/usado por vários utilizadores ao mesmo tempo.
- Pró-activo/Personalizado: O ecrã pode reagir aos utilizadores do sistema sem que para tal seja necessária a interacção directa destes. Com a detecção e reconhecimento dos utilizadores o ecrã adapta a informação a ser visualizada.
- Interactivo: Os utilizadores podem interagir com o ecrã.

Adicionalmente, apresentam-se alguns pontos referentes a questões que se relacionam com a utilização de ecrãs, independentemente da tecnologia utilizada. Os pontos são os seguintes:

- Noção do contexto: Para que um sistema ubíquo, que utilize ecrãs, funcione bem terá que saber onde se encontra o utilizador e, em determinados casos, qual a sua actividade.

Assim sendo, a utilização, seja no ambiente ou nos próprios ecrãs, de tecnologias com capacidade sensorial constitui a melhor solução para obter dados sobre o contexto. Podem ser utilizados sistemas tipo projector-câmara orientável, placas de sensores ou RFID, entre outros.

- **Privacidade:** Esta questão será, porventura, a mais delicada. Com a utilização de ecrãs que se pretendem públicos, existe a possibilidade de informação considerada sensível ficar visível para vários utilizadores. Existem factores sociais, como o embaraço em público, que limitam os utilizadores [19]. Se o utilizador não tem controlo em relação ao dispositivo de visualização, pode acontecer que o ecrã usado seja o errado no momento errado. Consequentemente, a noção do contexto torna-se essencial para a determinação da informação a ser apresentada. Haverá sistemas ou aplicações em que o utilizador terá que controlar a informação e o ecrã a utilizar.
- **Multi-interface multi-utilizador:** Para um utilizador, dependendo da aplicação, a informação poderá deixar de estar concentrada num único ecrã, podendo encontrar-se distribuída pelo ambiente. O facto de poderem existir grandes ecrãs permite um outro tipo de trabalho colaborativo com vários utilizadores a interagirem com a mesma superfície de visualização, ou, simultaneamente, com a mesma aplicação, mas através de vários ecrãs. Torna-se evidente que tenham que surgir novas abordagens no desenho das interfaces.
- **Suporte aplicacional efectivo:** Será necessário que se adopte um novo conceito no desenvolvimento de aplicações, uma vez que a interface de um utilizador com o sistema pode encontrar-se distribuída por vários ecrãs. As aplicações devem determinar que ecrãs se encontram disponíveis e a localização dos utilizadores relativamente aos ecrãs, entre outras condicionantes que estarão dependentes do tipo de sistema a implementar.
- **Noção de proprietário:** Apesar de na maior parte dos ambientes um utilizador não necessitar de ser "proprietário" de um dos ecrãs, poderão existir aplicações que obriguem a tal, nem que seja por breves instantes. Caso dois utilizadores se aproximam de um ecrã onde deve aparecer informação personalizada, como poderão ser resolvidos certos conflitos de disputa? Deve a disputa ser mediada em função dum tempo limite de uso ou da proximidade? Estas e outras questões colocam-se quando ecrãs, e principalmente grandes ecrãs, são utilizados enquanto recursos partilhados num ambiente de Ubicomp.

### 4.1.4 Informação Contextual Embutida

Hoje em dia, todas as pessoas têm, potencialmente, acesso a grandes quantidades de informação. Muita da informação disponível é contudo irrelevante para um qualquer utilizador em muitas das situações do seu dia-a-dia.

A interpretação do contexto por parte das aplicações já não é um conceito novo, mas ganhou especial interesse com o desenvolvimento dos novos dispositivos móveis, sendo muita da investigação que se faz direccionada para estes.

Contudo, a infra-estrutura baseia-se numa outra aproximação para a disponibilização de informação contextual, ou seja, esta deve ser apresentada, integrada, no espaço do objecto através dos ecrãs situados. Um ambiente poderá conter apenas um ecrã ou até um grande número destes dispositivos distribuídos pelo ambiente e junto aos respectivos objectos, ficando casualmente disponíveis para os utilizadores do ambiente. Pretende-se assim disponibilizar informação contextual junto a pontos de decisão que, na perspectiva do presente trabalho, são identificados pelos objectos que se pretendem aumentar. Este processo pode ser visto, simplesmente, como uma simples obtenção de mais e melhor informação sobre o objecto, mas também como uma ajuda para que se tomem melhores decisões relativamente às actividades a desempenhar.

Assim sendo, considera-se que aumentar objectos através de ecrãs que disponibilizam informação adicional é providenciar informação integrada no próprio contexto. Esta opção não impede que os ecrãs dos dispositivos móveis sejam usados, pois podem encontrar-se embutidos no contexto. Seguindo este princípio, é fundamental que se refiram os seguintes critérios [24]:

1. Embeber informação onde e quando é útil. É essencial que a informação apareça junto aos objectos e no momento em que os utilizadores se "cruzam" com os mesmos. É neste contexto que poderá ser útil, não se perdendo a ligação entre objecto e informação (no ecrã). Em cenários nos quais o objecto é uma instituição, é importante a colocação do ecrã num ponto vital, ajudando a que o objecto seja realmente aumentado com informação útil e imediata para os utilizadores.
2. Embeber a informação de uma forma o menos intrusiva possível. Em cenários nos quais já existam ecrãs dever-se-á aproveitá-los não alterando o ambiente encontrado. Nos restantes cenários há que integrá-los tentando fazer com que não sejam um factor de descontextualização. Apesar de os ecrãs já serem, só por si, um factor de ruído poderão ser integrados de forma suave, disponibilizando informação de forma agradável, simples e directa.
3. Providenciar informação sem que seja necessária interacção por parte do utilizador. À

partida, pode-se considerar a interacção como passiva pois apenas será necessária a co-presença de utilizador e objecto/ecrã.

#### 4.1.5 Ubiquidade dos Dispositivos Móveis

Os dispositivos móveis pessoais, em particular os telemóveis, estão a tornar-se em dispositivos computacionais ubíquos ao encontrarem-se em praticamente todos os locais, adaptando-se individualmente, e estando quase que universalmente conectáveis (com capacidades de comunicação tanto para curtas como para grandes distâncias). Ao serem constantemente transportados pelos seus utilizadores (tornaram-se ferramentas essenciais para o seu dia-a-dia), acabam por estar presentes quando ocorrem interacções com objectos em possíveis ambientes de UbiComp. Em paralelo com a evolução das funcionalidades disponíveis, existe um incremento na diversidade e capacidade das várias tecnologias, com especial destaque para a comunicação *wireless* (ver Subsecção 3.6).

A infra-estrutura proposta deverá assim ter em consideração a ubiquidade que os dispositivos móveis demonstram possuir. Dependendo do contexto e tipo de espaço físico para implementação, os dispositivos poderão ser uma mais valia caso sejam utilizados para a detecção do utilizador.

Deste modo, o espaço físico pode sentir a co-presença de um utilizador de uma forma quase natural e não intrusiva, no sentido em que o utilizador não necessitará de levar consigo um identificador extra para ser reconhecido no espaço aumentado. Para este efeito, parte-se do princípio que existe uma relação de um para um entre um utilizador e o seu dispositivo móvel pessoal. Sendo assim, um objecto/ecrã pode detectar a co-presença de um dispositivo com o qual se "cruza", bastando que implemente, preferencialmente, uma das tecnologias de comunicação *wireless* de curto alcance (e.g., Bluetooth) disponíveis nos dispositivos. Numa implementação deste tipo o sistema poderá detectar tanto utilizadores familiares, que são identificados por se encontrarem previamente registados, como utilizadores estranhos, que podem ser transeuntes detectados mas não identificados. São vários os projectos de investigação recentes que têm adoptado um cenário deste género, com interacções entre dispositivos móveis e ecrãs situados, sendo [145] um dos primeiros a fazê-lo.

Por outro lado, os dispositivos poderão ser utilizados como ferramentas que adicionam, à infra-estrutura, o tipo de interacção activa e mais personalizada.

Sabendo-se que os ecrãs situados são uma aposta em muitos sistemas de UbiComp, há quem prefira utilizar os pequenos ecrãs dos dispositivos móveis pela mobilidade e maior flexibilidade na

personalização que apresentam [17, 18]. Enquanto que o conceito de disponibilização automática de informação contextual em ecrãs situados pode ser alcançado exigindo pouco dos utilizadores (interacção passiva), já a interacção activa e personalizada requer um esforço técnico adicional por parte destes (i.e., com configurações e autenticações). Contudo, os dispositivos móveis poderão complementar o serviço oferecido pelo ecrã situado. Um objecto pode ser aumentado através da conjugação dos ecrãs públicos maiores com os ecrãs privados menores. Este conceito permite, não só que informação pública personalizada possa ser providenciada como, abordar conteúdos considerados privados que relacionam o utilizador com o objecto aumentado. Existe ainda a vantagem adicional de o dispositivo móvel poder ser o responsável pelo início da interacção activa. Para tal, devem ser considerados os sensores (câmara, microfone e interfaces de comunicação) actualmente integrados nos dispositivos móveis modernos [146].

Assim sendo, a arquitectura da infra-estrutura apresentará um módulo para interacções activas, aproveitando a ubiquidade dos dispositivos móveis pessoais.

## 4.2 Arquitectura da infra-estrutura

Nesta secção, desenha-se a arquitectura genérica para implementação dos sistemas pretendidos, mas primeiro resumem-se os requisitos essenciais aos quais deve obedecer.

### 4.2.1 Requisitos

Pretende-se que a proposta de arquitectura leve à fácil e rápida implementação de sistemas que aumentem objectos físicos. Além de considerar aos princípios apresentados para a infra-estrutura, deverá ter em consideração os requisitos seguintes:

- Considerar tecnologias estáveis de uso comum e/ou fácil integração nas actividades diárias para a detecção da co-presença dos utilizadores.
- Permitir que a informação adicional resulte no surgimento de serviços adicionais relacionados com os objectos.
- Integrar uma configuração adicional opcional referente à interacção activa, na qual o utilizador poderá fazer opções que determinem a informação a mostrar/receber ou a activação de serviços.
- Distribuir o processamento pelas várias entidades presentes no ambiente, i.e., pelos utilizadores/dispositivos, objectos/ecrãs e servidores.

- Providenciar ao programador abstrações que escondam os detalhes das tecnologias de comunicação usadas entre o utilizador/dispositivo móvel e o objecto/ecrã.
- Providenciar ao programador abstrações que escondam os detalhes da implementação do sistema de visualização.
- Providenciar um componente que faça a gestão do contexto em que decorre uma interacção.
- Providenciar componentes, para o dispositivo móvel, que permitam o fácil desenvolvimento de aplicações clientes leves quanto a exigências de memória e processamento.

### 4.2.2 Arquitectura Genérica

A presente secção é dedicada à proposta de uma arquitectura genérica e modular. A arquitectura deve ser abrangente para que englobe vários tipos de cenários e se consiga assim diminuir o tempo de implementação dos sistemas.

A implementação de um sistema pervasivo para aumentar objectos físicos com informação adicional contextual envolve múltiplos componentes e apresenta vários problemas. Dois exemplos de arquitecturas interessantes, que se relacionam com o trabalho presente, são: 1) uma aproximação direccionada para as interacções dos dispositivos móveis com os objectos [10], e 2) uma implementação de um modelo conceptual centrado nas actividades encontradas num ambiente [133].

A elaboração de uma arquitectura genérica por módulos permite maximizar a sua aplicação em cenários diferentes. Esta abordagem permite perceber como os vários componentes se podem e devem interligar, adequando a arquitectura às necessidades de cada aplicação/projecto. A separação modular apresenta as seguintes características [53]:

- Partilha de processamento - A possibilidade de divisão do processamento por vários dispositivos auxilia na determinação da quantidade de informação a ser processada pelos múltiplos módulos. Deste modo, poderá haver um equilíbrio entre a capacidade de processamento que é disponibilizada individualmente por cada dispositivo e a quantidade de informação a ser processada. É de salientar que a distribuição é efectuada à custa de meios de comunicação entre os diferentes módulos, sendo um factor de introdução de tempos de atraso adicionais. A divisão do processamento, principalmente ao nível de grandes quantidades de informação, só será vantajosa se houver um grande equilíbrio do processamento e se forem considerados os tempos de trânsito da informação.
- Facilidade de implementação - Esta opção garante que cada módulo seja responsável por um problema mais pequeno. É possível ter um núcleo inicial em funcionamento ao qual

podem ser adicionados módulos que emprestam novas funcionalidades ao sistema.

- **Reutilização** - É fundamental que a arquitectura tenha em consideração esta característica elementar num sistema modular. A infra-estrutura apresenta um conjunto de módulos desenvolvidos que podem ser reutilizados ou adaptados na implementação de novos sistemas, que possuem uma base idêntica.
- **Adaptável** - Alguns dos módulos não são considerados essenciais, não fazendo parte do núcleo da arquitectura. Assim sendo, dependendo da aplicação pretendida, pode facilmente adaptar-se a arquitectura através da adição ou remoção de módulos.
- **Sincronização** - Pelo facto de existirem módulos a serem executados em diferentes dispositivos, ou em plataformas de processos paralelos, é necessário garantir a coerência e a sincronização entre os múltiplos módulos. Esta desvantagem que causa atrasos pode, por vezes, ser minimizada caso não exista necessidade de sincronização entre módulos.
- **Tempos de latência** - Com os módulos surgem também os tempos de latência devido à comunicação entre dispositivos. Factores como os protocolos, a rede, e até mesmo a distância física entre componentes, são determinantes para este ponto.
- **Protocolos** - A uniformização da comunicação entre diferentes módulos deve passar por um processo de definição de protocolos. Uma correcta definição de protocolos para os módulos independentes permite uma utilização mais fácil e rápida por parte de qualquer aplicação.

A arquitectura proposta assenta em três grandes componentes, que podem ser vistos como módulos genéricos e independentes, correspondendo a três camadas de implementação (ver Figura 4.2). Os componentes são os seguintes:

- **Rastreo:** Em qualquer aplicação deste tipo tem que existir um mecanismo de localização e identificação do utilizador/dispositivo (etiqueta). É este o responsável por sentir a co-presença de um utilizador na área do respectivo objecto. Adicionalmente, este componente pode ser estendido para integrar uma interface de comunicação de curto alcance com o utilizador.
- **Núcleo Aplicacional:** Este é o componente responsável pela implementação da aplicação. Deve ser capaz de analisar e processar a informação dada pelo módulo acima descrito, cruzando-a com a informação contextual referente ao objecto. Aqui reside a independência da aplicação, que depende do cenário e dos objectivos. Este componente inclui os módulos seguintes: Gestor de Informação, Personalização e Conteúdos de Objectos.
- **Interface Ecrã:** Por último, a apresentação/visualização da informação adicional tendo

em consideração o estímulo principal, a co-presença do utilizador. Nesta dissertação, consideraram-se os ecrãs (cuja tecnologia utilizada varia de acordo com a implementação) como sendo suficientemente versáteis para a apresentação da informação.

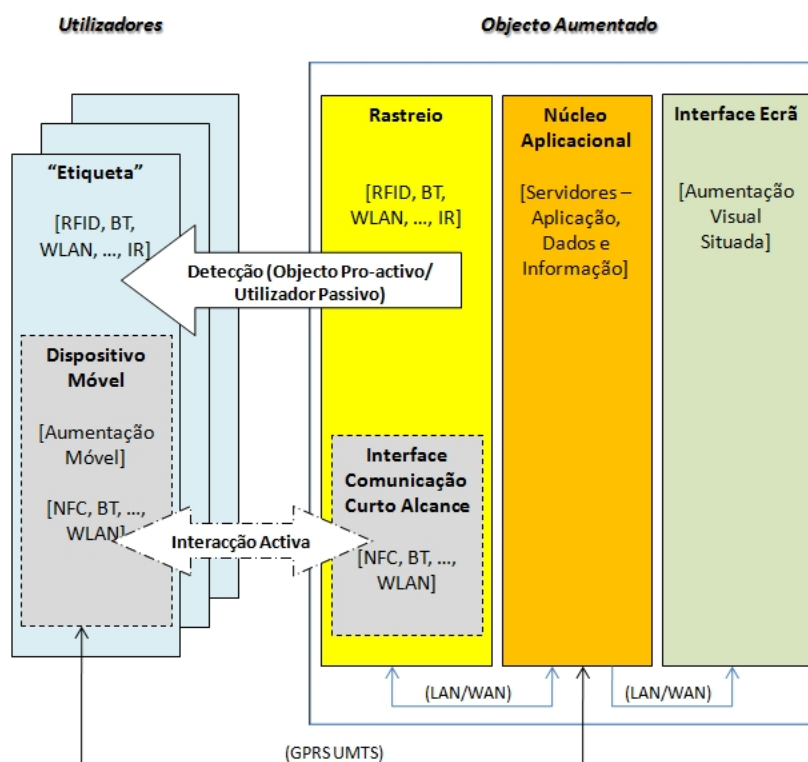


Figura 4.2: Arquitectura Genérica da Infra-estrutura.

Os três componentes referidos encontram-se na camada correspondente ao objecto aumentado tendo em vista o princípio fundamental de pró-actividade do par objecto/ecrã em relação ao utilizador. Este último pode ser visto como uma simples etiqueta, por exemplo, RFID ou telemóvel com Bluetooth activado, que tem uma interacção passiva resultante da sua presença natural no cenário. Adicionalmente, em certas aplicações pode existir a vontade em dotar o utilizador de um papel activo. A arquitectura prevê a interacção activa com um componente para aumentar o objecto através de dispositivo móvel. Deste modo, além de se aumentar o objecto através do ecrã grande situado, o componente de rastreio apresenta uma extensão para o aumentar através de comunicação com o utilizador, que já não é visto como uma etiqueta simples. A utilização de um dispositivo móvel, que pode ser pessoal, permite a comunicação com o objecto através de tecnologias de curto alcance ou através de GPRS/UMTS. Permite ainda a visualização de informação adicional no ecrã privado móvel do mesmo, sendo um complemento ao ecrã público situado. A distribuição dos componentes pelas diferentes máquinas/dispositivos

existentes depende exclusivamente do poder computacional destas, e a forma de equilibrar o processamento necessário depende muito da capacidade dos recursos disponíveis.

A proposta também pode ser vista como uma típica arquitectura cliente/servidor, considerando um conjunto de servidores dedicados a fornecer informação a vários clientes. O conjunto formado pelo utilizador e o objecto pode ser visto como cliente. Tanto o componente para rastreio como o de interface encontram-se junto ao objecto, ou seja, do lado do cliente. A presença do utilizador junto do objecto é detectada pelo componente de rastreio que solicita um serviço ao núcleo aplicacional, o servidor. Este envia os resultados para o módulo Interface Ecrã. Também pode comunicar com o mecanismo de rastreio, e até mesmo com o dispositivo móvel do utilizador. Dependendo da aplicação, o núcleo aplicacional pode ser partilhado por vários clientes (objectos) ou estar dedicado apenas a um objecto cliente, tendo este uma extensão por vários utilizadores com dispositivos móveis.

Uma vez que existem vários módulos/componentes a comunicarem entre si, podendo ser remotos (encontrando-se em máquinas/dispositivos separados), torna-se necessária a existência de uma interface de comunicações (que integre, por exemplo, NFC, BT, GPRS e/ou WAN/LAN) capaz de suportar de uma forma transparente e eficiente a comunicação de dados entre esses módulos.

### 4.2.2.1 Os Módulos Centrais da Arquitectura

O processamento mais complexo é efectuado por um módulo de gestão de informação, que, na essência, após receber uma entrada vinda do módulo de rastreio, activa um módulo interpretador contextual que indica como a informação adicional deve ser composta. São questionados dois serviços de dados distintos com vista a obter a informação adicional total. A informação é então direccionada para o objecto/ecrã ou cliente móvel que a solicitou. A Figura 4.3 apresenta uma arquitectura mais detalhada, onde são visíveis os módulos principais que a integram. Estes módulos correspondem à camada do objecto aumentado, implementando apenas o requisito primário de interacção passiva.

Passa-se a descrever mais detalhadamente cada um dos módulos e sub-módulos determinantes.

- **Rastreio:** Este módulo é o complemento para a aumentação visual do objecto, sendo essencial para qualquer aplicação na área. É o responsável pela detecção e identificação do utilizador, adicionando ao objecto a capacidade de percepção de co-presença. É, por

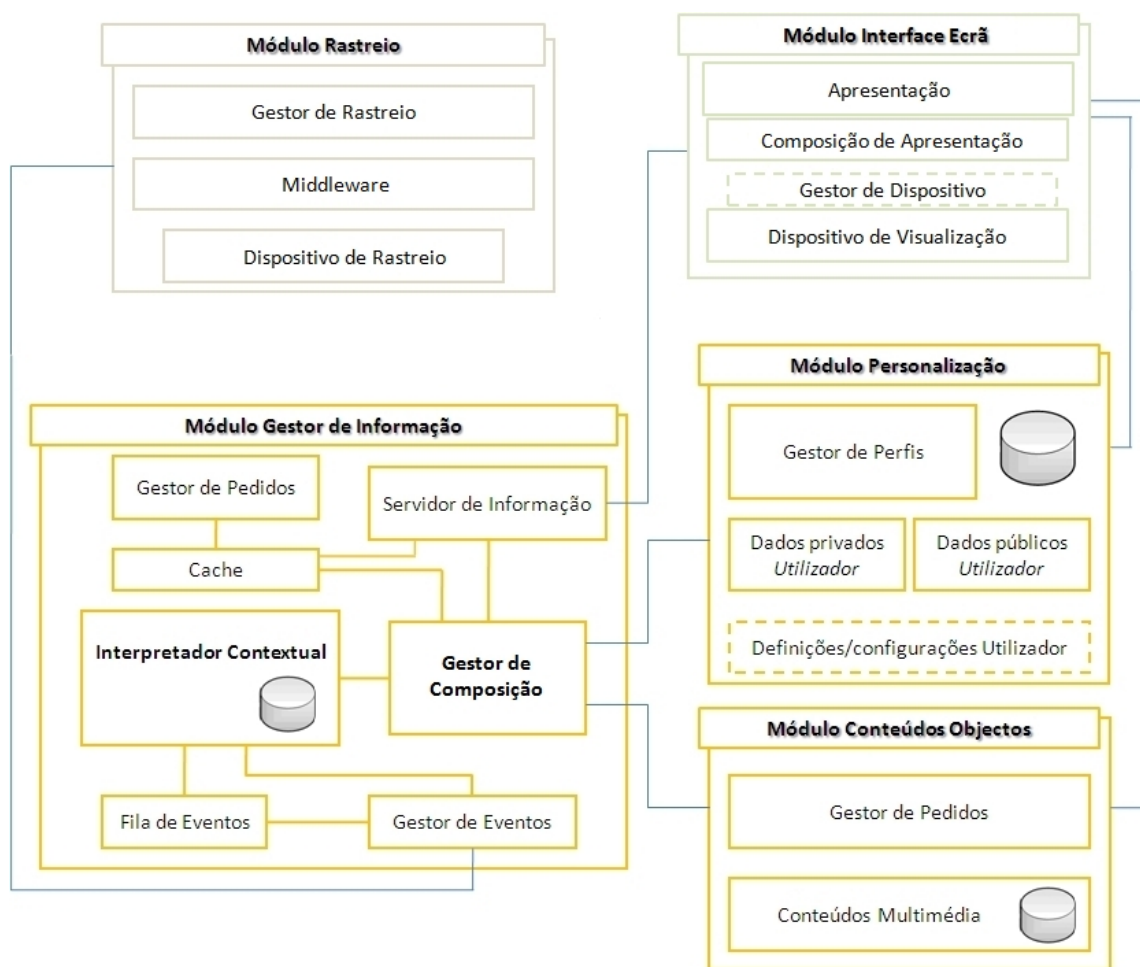


Figura 4.3: Módulos da Arquitectura Genérica (Objecto Aumentado).

exemplo, através deste módulo que se pode saber durante quanto tempo o utilizador se de-  
teve na área do objecto. Num espaço repleto de objectos aumentados consegue-se conhecer  
o percurso do utilizador ao longo dos objectos, conhecendo-se a localização e podendo-se  
estimar o sentido da movimentação. São várias as tecnologias que podem ser utilizadas  
neste módulo (e.g., RFID, Bluetooth, etc. - ver Subsecção 3.3), sendo aconselhável criar  
uma camada de abstracção - **Gestor de Rastreio** - que funcione como interface de saída  
e providencie de forma linear todos os dados ao núcleo aplicacional. Destaque ainda para  
a camada **Middleware** que se encontra dependente da tecnologia utilizada. Esta camada  
faz a abstracção do **Dispositivo de Rastreio** (*hardware*) utilizado, sendo essencial para  
a camada superior, o Gestor de Rastreio, que deste modo apenas tem que integrar a(s)  
interface(s) disponibilizadas pelo Middleware.

- **Interface Ecrã:** É o módulo saída do sistema implementado, sendo responsável pela

**Apresentação** da informação adicional sobre o objecto e de possíveis serviços. A informação deve ser apresentada o mais rapidamente possível ao utilizador, verificando-se o menor tempo de latência entre o momento de detecção/identificação do utilizador e o de visualização da informação. Uma vez que a informação adicional é direccionada para o utilizador detectado, não fará muito sentido que este já não se encontre no local quando a mesma for apresentada. O cenário de aplicação, e em particular a cadência de utilizadores, é determinante para a escolha da tecnologia a usar. Visto existir uma grande diversidade nesta escolha (ver Subsecção 3.7), poderá existir a necessidade de se criar uma camada de gestão/abstracção - **Gestor de Dispositivo** - para determinados casos de uso, i.e., criar protocolos uniformes para gestão do(s) **Dispositivo(s) de Visualização**, facilitando a programação do sub-módulo **Composição de Apresentação**. Este módulo questiona e recebe mensagens do servidor no módulo central da aplicação. As mensagens dão indicações sobre os intervenientes na interacção, i.e., objecto, utilizador e respectivo perfil, e ainda sobre o tipo de informação adicional. Este tipo indica que tipo de interface gráfica deve ser construída/utilizada, enquanto que os módulos de Personalização e de Conteúdos de Objectos devem ser questionados para que se obtenham os conteúdos para a apresentação.

- **Personalização:** Este módulo comporta-se como um serviço de dados que faz a gestão dos perfis da aplicação e dos dados dos utilizadores, podendo integrar um sub-módulo opcional para definições individuais. O sub-módulo **Gestão de Perfis** encarrega-se da definição e da gestão de um conjunto de perfis baseados em preferências e interesses relacionados com o cenário de aplicação. Uma vez que a quantidade de informação adicional sobre um objecto pode ser muita, justifica-se que a mesma seja estruturada e organizada de acordo com os perfis criados para o cenário. Num sistema de partilha de ecrã público este é um requisito essencial. Da mesma forma, o utilizador não deverá receber toda a informação sobre o objecto no seu dispositivo móvel, em caso de cenário com interacção activa. Nem toda a informação é relevante para um determinado utilizador. É ainda este módulo o responsável pelo repositório dos dados do utilizador. Estes estão separados em **Dados públicos** e em **Dados privados** do utilizador. Os primeiros podem ser livremente relacionados com a informação do objecto e/ou apresentados directamente no ecrã público. Já os segundos são resguardados e mantidos para conhecimento do sistema, podendo ser disponibilizados em cenários que envolvam a interacção activa do utilizador através de dispositivos móveis pessoais. Pode existir um sub-módulo adicional de **Definições/Configurações de**

**Utilizador** que em determinadas aplicações pode ser útil. Assim sendo, configurações, calibrações e definições de preferências individuais são armazenadas para posteriormente ser mais simples a configuração automática do sistema para cada utilizador. Não sendo visto como algo essencial para todas as aplicações, pode ser interessante sempre que se quiser adaptar os módulos de rastreio e interface ecrã aos utilizadores.

- **Conteúdos de Objectos:** Este serviço de dados é responsável pelo repositório de dados sobre os objectos presentes no cenário. O repositório é tipicamente extenso e volumoso, incluindo os mais variados conteúdos multimédia, tais como texto, som, imagem e vídeo. Este módulo é questionado pelos módulos Gestor de Informação e Interface Ecrã. O último exige um maior tráfego de dados visto ser o responsável pela composição final da apresentação da informação ao utilizador. Já o primeiro faz questões nas quais pede basicamente texto e imagens para compor mensagens com a informação sobre o objecto. As respostas a estes pedidos devem ser rápidas, pois a informação criada é disponibilizada num servidor para ser acedida posteriormente pelo Interface ecrã ou por um módulo presente num dispositivo móvel. Caso seja para o ecrã, então depois o respectivo sub-módulo Composição de Apresentação voltará a questionar este módulo, para obter o maior volume de conteúdos, se for necessário para a apresentação em questão. Já o dispositivo móvel não obrigará, em geral, a uma segunda consulta deste módulo. É necessária uma camada de gestão - **Gestor de Pedidos** - para resolver eventuais conflitos e gerir a concorrência em cenários que englobem vários objectos e muitos utilizadores.
- **Gestor de Informação:** A arquitectura do sistema apresenta uma orientação para o evento, na medida em que cada vez que o utilizador altera o seu estado, i.e., sendo, ou deixando de ser, detectado junto a um objecto/ecrã, funciona como um evento para o qual existe um *handler*, garantindo-se a actualização do estado da aplicação. O sistema reage a estímulos externos, as interações dos utilizadores, mantendo-se estável quando esses estímulos não ocorrem. Assim sendo, a porta de entrada principal deste módulo é o **Gestor de Eventos**, que controla as entradas geradas pelo módulo de rastreio. A arquitectura aconselha ainda a utilização de uma **Fila de Eventos**, para garantir que estes não se perdem e possibilitando, ao mesmo tempo, a programação de situações temporais. O núcleo deste módulo reside nos sub-módulos **Interpretador Contextual (IC)** e **Gestor de Composição (GdC)**. Com a obtenção da informação do módulo de rastreio, o Interpretador é responsável por correlacionar o utilizador detectado, que possui um de-

terminado perfil, com o objecto que o detectou para que posteriormente se obtenha a informação adicional correspondente. No passo seguinte, envia uma mensagem (documento) para o GdC dando conhecimento sobre a interpretação realizada. O sub-módulo de composição combina essa informação com a que recolhe dos servidores de dados (Personalização e Conteúdos de Objectos), produzindo um outro documento para o **Servidor de Informação**. O documento é também produzido de acordo com a origem da detecção ou do pedido. Caso o GdC tenha recebido a mensagem da parte do Interpretador, então o documento produzido deve conter um mínimo de informação, pois é direccionado para o módulo Interface Ecrã, que então o utiliza para saber como compor a apresentação multimédia final. Se o GdC receber um pedido vindo de um dispositivo móvel (uma interacção activa), então o documento a disponibilizar no servidor deve ser o final, contendo todos os dados questionados ou esperados. Dependendo do cenário, o GdC pode ter ainda em conta, por exemplo, as definições locais, as preferências do utilizador, e a localização do objecto. Este módulo de composição é o responsável por compor, estruturar e seleccionar a informação que deve ser apresentada.

Se, por um lado, o sub-módulo GdC está dependente da aplicação, já o sub-módulo IC é completamente independente, apresentando uma arquitectura (modelo) que se adequa a qualquer cenário (ver Figura 4.4). O modelo funciona caso se esteja perante um cenário composto por vários objectos, como por exemplo numa sala de um museu, mas também nos casos em que se considera apenas um grande objecto conceptual, como por exemplo uma instituição. Permite que possam ser associados vários ecrãs ao objecto, sendo colocados em várias localizações da instituição. Um "ObjectoAumentado" corresponde a um "Objecto", a uma "Localização", e tem associados um "Leitor" (que pode ser apenas uma antena, dependendo do tipo de tecnologia utilizada) e um "Ecrã". Já a "InformaçãoAdicional" indica que dados de um "Objecto" devem ser conjugados tendo em consideração um determinado "Perfil". O "TipoInfo" é importante para o GdC, pois indica quais os tipos de informação multimédia que se encontram presentes na porção de "InformaçãoAdicional", condicionando a interface visual da apresentação. Um "UtilizadorIdentificado" corresponde a um "Utilizador" com uma determinada "Etiqueta" (e.g., uma *tag* de RFID ou um dispositivo móvel com Bluetooth) durante um determinado período de tempo, que pode ser menor que uma hora ou durar muitos meses. Um "Utilizador" tem sempre um "Perfil", nem que seja o mais genérico, caso tenha uma actividade residual no cenário de aplicação. Dependendo do tipo de "Etiqueta" (tipo de tecnologia) utilizada, são vários os

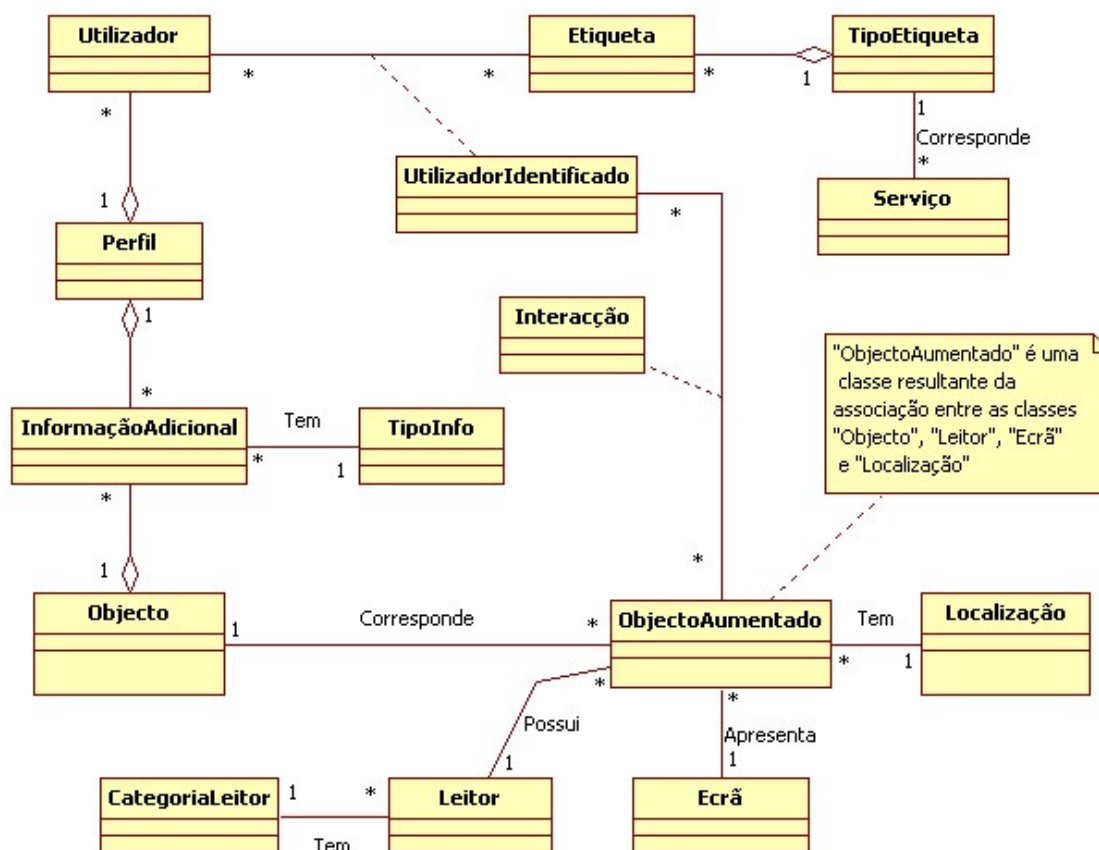


Figura 4.4: Arquitectura do Interpretador Contextual.

serviços que podem ser disponibilizados, podendo ser mais preciosos num contexto que permita a interação activa por parte do utilizador. Por fim, sempre que um utilizador é detectado por um "ObjectoAumentado", é criada uma nova "Interação" que, além de disparar um novo evento, regista vários dados que ajudam a perceber o tipo de interação. Assim sendo, este sub-módulo é também um serviço de dados que contém a informação elementar sobre os objectos, os perfis, os utilizadores, os dispositivos de identificação e a informação adicional de modo a poder relacionar todas estas entidades, guardando ainda todas as interações detectadas entre utilizadores e objectos.

O Servidor de Informação disponibiliza os documentos finais, criados pelo GdC, tanto para o módulo Interface Ecrã como para os pedidos vindos dos dispositivos móveis, que são utilizados por utilizadores para interações activas. O mecanismo de **Cache** é implementado para garantir uma maior robustez e uma maior rapidez do sistema. Integra-se um sistema assíncrono de pedidos de informação ao Gestor de Informação, para que não seja necessário

parar o processamento enquanto se espera pelo resultado dos pedidos ao Servidor ou ao GdC. Dependendo do cenário de aplicação, e dos serviços disponibilizados na interacção activa, um pedido pode ter de passar primeiro pelo GdC, caso se tenha que produzir um novo documento de informação. O **Gestor de Pedidos** é o responsável pela informação passada ao Módulo Rastreio, que realiza a comunicação de curto alcance com o dispositivo móvel. Por outro lado, um pedido pode ir directamente para o Servidor quando é referente a informação anteriormente gerada, sendo disponibilizada via GPRS.

### 4.2.2.2 Os Módulos para Aumentação Móvel

A arquitectura genérica apresenta ainda uma camada adicional no lado do utilizador (ver Figura 4.2). Esta pode ser implementada sempre que sejam utilizados dispositivos móveis (e.g., telemóveis, PDAs) nas aplicações. Num cenário que tenha em consideração o dispositivo móvel, como etiqueta que identifique o utilizador, torna-se natural utilizá-lo como ferramenta do utilizador para interacção activa.

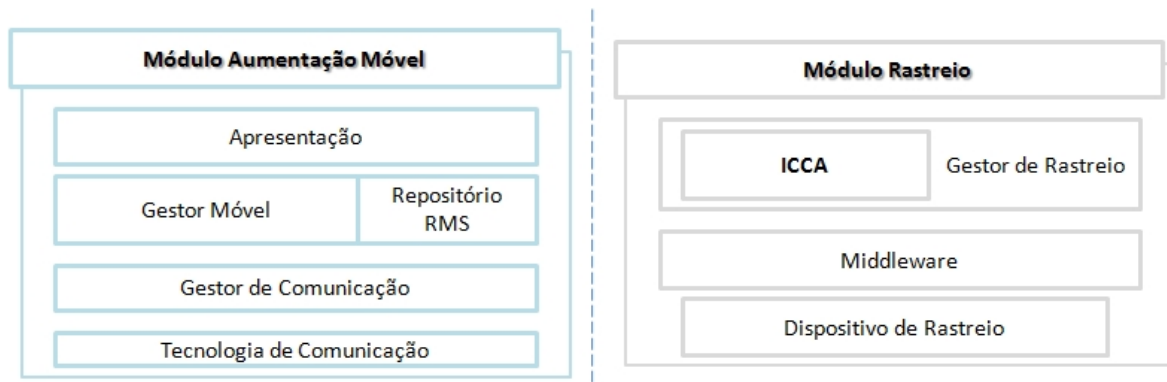


Figura 4.5: Módulos adicionais da arquitectura: (esquerda) Módulo do dispositivo móvel e (direita) adição do ICCA ao módulo Rastreio.

A Figura 4.5 apresenta à esquerda o módulo **Aumentação Móvel**, que proporciona ao utilizador uma aumento adicional do objecto, podendo ir além do ambiente de aplicação. Para tal, permite a comunicação directa, através de uma tecnologia de curto alcance, com o módulo de Rastreio e a comunicação à distância, fazendo pedidos, com o Gestor de Informação do sistema. Este módulo assenta num **Gestor de Comunicação** que providencia uma abstracção quanto à tecnologia concreta de ligação utilizada (e.g., NFC, Bluetooth, GPRS). Desta forma, torna a programação do par **Gestor Móvel/Apresentação** (a aplicação) menos complexa e independente da tecnologia a usar, que, ao longo do tempo, até pode variar sem que para tal

obrigue a uma reprogramação da aplicação. O **Gestor Móvel** integra um sub-módulo opcional para gestão de serviços, estabelecendo as facilidades colocadas ao dispor do utilizador no processo de aumento do objecto, encontrando-se fortemente ligado à tecnologia utilizada. É ainda integrado um sub-módulo chamado **Repositório RMS** (*Record Management System*), uma vez que a informação adicional relacionada com a contextualização do utilizador e do objecto pode ser guardada para permitir uma visualização posterior ao utilizador. A **Apresentação** está fortemente dependente das capacidades do dispositivo em termos de ecrã e da aplicação implementada.

Do lado do objecto, para que exista a possibilidade de interacção directa e próxima é necessário que se adicione no módulo de rastreio um novo sub-módulo integrado no **Gestor de Rastreio**. Designou-se por **ICCA** (Interface de Comunicação de Curto Alcance) e permite dotar o Gestor, que já funcionava como camada de abstracção e disponibilização de serviços, de capacidade para estar atento a dispositivos que com ele queiram comunicar. Sempre que o módulo de rastreio detectar um utilizador nas imediações do objecto gera um evento de detecção para ser capturado pelo Gestor de Eventos da aplicação. Contudo, sempre que for o dispositivo móvel (a etiqueta) a desencadear a comunicação é o ICCA quem assume o controlo, acabando por estabelecer uma ligação com o Gestor de Pedidos da aplicação. O próprio ICCA, ao detectar um utilizador, pode questionar o Gestor, para obter a informação privada recente, sem que o dispositivo o tenha solicitado e para comunicar depois com este último. Como alternativa à interacção activa próxima, e à comunicação com o módulo de rastreio, existe a possibilidade de o dispositivo fazer pedidos directos à aplicação (e.g., através de GPRS), que disponibiliza no Servidor de Informação os documentos com a informação adicional.



## Capítulo 5

# Realização do Protótipo divingForPearls

Este capítulo é dedicado à apresentação do primeiro protótipo de realização da infra-estrutura proposta, que se centra numa actividade distribuída por vários objectos/ecrãs.

O projecto divingForPearls visa aplicar e testar a infra-estrutura num cenário do tipo museu (ver Secção 5.1). Até ao momento, o protótipo foi testado no interior do centro de investigação RSC (RFID Solutions Center) da Sybase Portugal<sup>1</sup>. A tecnologia de rastreio escolhida foi a RFID, enquanto que para o desenvolvimento dos módulos de *software* a linguagem C#, na plataforma .Net, foi a escolha central tendo em conta a facilidade de integração com as escolhas efectuadas ao nível de RFID.

### 5.1 Tipo de Cenário e Actividades

São várias as actividades que decorrem em museus, ou noutros ambientes similares, e que poderiam beneficiar da implementação de um sistema que aumentasse as peças em exposição. Estas podem tornar-se inteligentes através da oferta de serviços e da noção da co-presença dos visitantes do espaço. Para os visitantes (utilizadores) será estimulante terem acesso a uma maior quantidade de informação sobre as peças sem que para tal tenham que desenvolver esforços suplementares. Sempre que um utilizador se aproxima de uma peça (objecto) para a observar existe uma interacção física com a mesma. Apesar de existir a intenção de aproximação ao objecto por parte do utilizador, a interacção pode ser considerada acidental. O utilizador tem em mente a observação do objecto, mas este "sente" a co-presença do utilizador podendo aproveitá-la

---

<sup>1</sup><http://www.sybase.pt>

para oferecer outro(s) serviço(s). O serviço essencial (central) na implementação será aumentar o objecto através da disponibilização de informação adicional contextual num ecrã que acompanhe o objecto.

Assim sendo, com um sistema deste tipo, os utilizadores podem receber mais informação sem que para tal exista uma interferência no procedimento vulgar de visita do museu. Não são necessárias acções adicionais por parte dos utilizadores que se podem limitar a percorrer o espaço e a observar os objectos. Na essência, existirá sempre a vantagem de o espaço continuar a ser um museu tradicional, caso o utilizador não queira utilizar o sistema.

Este conceito de visita, em que os objectos se aumentam, permite ao utilizador descobrir informação adicional que de outra forma não estaria disponível ou necessitaria de demasiados recursos e/ou acções para ser visualizada. Baseando-se nos sistemas de informação ambiental, o sistema do museu será o menos intrusivo possível, não desviando o utilizador da actividade considerada normal.

Hoje em dia, são muitos os museus que disponibilizam no seu sítio na Web uma visita virtual aos espaços, exposições e peças que os integram. Pode-se considerar que, para o sistema proposto, existe uma conjugação de uma visita virtual com a visita física. Deste modo, a informação virtual, que pode ser mais ou menos rica em multimédia, pode ser cruzada no espaço físico com a informação que o utilizador retira da observação do objecto. Sempre que o utilizador entrar no raio de detecção de um objecto, este poderá fornecer-lhe no respectivo ecrã público a informação adicional.

A informação adicional a ser apresentada pode ser genérica, caso se consiga detectar o utilizador, mas não identificar quem é em termos de gostos e preferências. Um critério para a disponibilização da informação pode passar pelo tempo que um utilizador dedica à visualização do objecto. Por exemplo, se o utilizador observa, durante algum tempo, uma determinada obra então pode-se partir do princípio que esta lhe suscita interesse particular, indicando uma situação clara para aumento do objecto observado. A informação adicional poderá conter detalhes sobre a vida do pintor, a história do quadro, e como se enquadra num estilo. Assim sendo, um objecto apenas se aumentará para utilizadores que se detenham na sua zona por mais do que um determinado período de tempo.

Contudo, caso o utilizador seja conhecido, após registo no sistema/museu, então a informação adicional também poderá ser personalizada. Isto é, dependendo do objecto e da quantidade de informação adicional existente sobre o mesmo, as preferências e/ou perfil do utilizador detectado servirão para filtrar e adequar os conteúdos a serem visualizados.

À entrada do museu, de uma exposição ou de uma sala, dependendo do grau e tipo de implementação do sistema, poderá existir uma zona de registo para os visitantes. Um visitante pode desejar não fazer parte do sistema e assim não efectuar o registo, ou, estando registado, pode sempre reverter para a configuração normal, deixando de utilizar a etiqueta que faz com que seja detectado. No momento de registo pode ser atribuída, por exemplo, uma etiqueta de RFID ou Bluetooth. Na configuração básica e essencial do sistema, esta será para o visitante a única novidade relativa a uma visita normal, não exigindo que o utilizador tenha que utilizar explicitamente outros equipamentos. A etiqueta poderá ser levada ao pescoço e no caso da tecnologia Bluetooth, a etiqueta até poderá ser o próprio dispositivo móvel, seja pessoal ou disponibilizado pelo museu. Um visitante poderá, por exemplo, escolher um de dois tipos de utilizador: (1) geral, em que este apenas é detectado pelos objectos; (2) personalizado, no qual se enquadra num determinado perfil de acordo com um conjunto de preferências.

Existe também a hipótese de se complementar a visualização no ecrã público com a utilização de um dispositivo móvel, como um PDA, exigindo uma interacção activa. Nas situações em que sejam muitos os utilizadores do espaço, e/ou em que faça sentido existir uma grande quantidade de informação por objecto, a disponibilização da informação, ou de parte, no dispositivo permitirá que, ao afastar-se do objecto, o utilizador continue a ter acesso a informação sobre o mesmo. Sempre que o dispositivo seja pessoal, existirá a vantagem de poder levar consigo para fora do museu a informação adicional dos objectos que visitou, mantendo assim um histórico personalizado da sua visita. Poderão ainda ser disponibilizados alguns serviços adicionais que estarão dependentes da interacção activa a partir do dispositivo.

Quanto a serviços adicionais oferecidos pelo sistema, podem encontrar-se facilmente algumas oportunidades. Imagine-se, por exemplo, um utilizador a percorrer uma exposição e a ter como guia o(s) artista(s) a que pertencem as peças. Este serviço de guia virtual, e apesar de ser mais indicado para cenários puros de realidade aumentada, apenas altera o conceito de escolha e apresentação da informação adicional dos objectos. O guia até poderia ser um crítico ou uma outra personagem relacionada com o objecto e que fosse de encontro ao perfil do utilizador. Estes guias podem ter uma, entre várias, personalidades possíveis, fazendo com que a abordagem na apresentação da informação possa variar. Ainda nesta perspectiva de serviço de guia virtual, pode existir um aconselhamento quanto aos objectos seguintes a serem visitados. Vários factores teriam que ser considerados na proposta, tais como: o tempo disponível do utilizador para a visita; o seu perfil; o tipo de exposição; os objectos já "visitados"; entre outros. O cruzamento destes dados aconselharia a uma determinada sequência na visita. Um objecto pode aconselhar o

utilizador a visitar outros objectos. Mais uma vez, para este tipo de serviço, o tipo de interacção do utilizador também não tem de ser activo. O ponto principal é a informação adicional, focada nas necessidades de cada visitante. Este serviço de guia virtual pode ser consideravelmente reduzido nos momentos em que co-existam muitos utilizadores. O objecto poderá apresentar no ecrã público, para cada utilizador, apenas uma linha de informação adicional e uma simples indicação com a identificação de outro objecto a visitar.

Um outro serviço interessante será a possibilidade de disponibilização de um histórico da visita, para o qual cada objecto visitado pelo utilizador pode contribuir com conteúdos. Assim sendo, após a visita o utilizador poderia receber um documento referente ao percurso que efectuou e com as informações adicionais e contextuais sobre cada objecto. Existem várias possibilidades de apresentação para o histórico, que funcionaria como diário automatizado para o utilizador. O documento poderia corresponder a um sítio na Web ou ser um álbum digital entregue em formato CD/DVD à saída do museu.

Nas visitas feitas em grupo e com guia físico, este será o utilizador do sistema, tendo para tal um determinado perfil. Deste modo, a informação adicional de cada objecto será relacionada com o guia e não com cada um dos visitantes que formam o grupo. Caso o serviço de guia virtual esteja activo será "controlado" pelo perfil do guia físico que vai interagindo com os vários objectos. Como é natural, o guia físico será o interlocutor entre os visitantes e o sistema.

Um serviço destes funciona ainda como mais valia para o museu pois permite-lhe saber facilmente algumas estatísticas, tais como o número de visitas por objecto, o tempo contínuo dedicado pelos utilizadores aos objectos e os percursos efectuados.

## 5.2 Enquadramento Organizacional

Apesar deste sistema poder ser implementado em qualquer ambiente semelhante a uma sala de museu, com exposições de peças sobretudo estáticas, é bom ter em perspectiva uma aplicação num espaço real. Uma vez que o Museu Nacional de Arte Antiga<sup>2</sup> (MNAA) é um dos parceiros previstos no projecto referido na secção 1.1, no qual se insere a investigação relativa ao presente trabalho, então considera-se uma futura aplicação neste ambiente.

O MNAA encontra-se instalado no palácio de Alvor, do século XVIII, em Lisboa. É considerado o principal museu do país e é um dos mais visitados, sendo efectivamente importante por reunir o mais significativo espólio da arte portuguesa dos séculos XIV a XIX. O acervo do museu

---

<sup>2</sup><http://www.mnarteantiga-ipmuseus.pt/>

compreende as Artes Plásticas (Pintura, Escultura, Desenho e Gravura) e as Artes Decorativas (Ourivesaria, Cerâmica, Têxteis e Mobiliário), dividindo-se por três pisos onde se encontram várias zonas/salas de exposição que albergam as várias colecções.

Numa análise estratégica de implementação do projecto, o seu carácter inovador é um ponto forte a considerar. Apresenta uma facilidade na utilização do sistema, ao não obrigar os utilizadores a terem um papel tecnologicamente activo, mas apenas alguma atenção quanto à posição escolhida junto às peças. Este sistema pode disponibilizar alguns serviços adicionais, incluindo aqueles que podem melhorar as questões de segurança. Esta evolução tecnológica, num ramo onde sistemas destes ainda são raros, pode cativar clientes, ao actualizar e melhorar a imagem para o exterior.

Existem alguns pontos que podem ser considerados negativos por uma instituição como o MNAA, tais como uma necessidade de investimento inicial relativamente elevado, os cuidados extra na preparação das colecções e na configuração das salas de exposição, ou a possível descharacterização da exposição com a introdução de objectos tecnológicos. Deste modo, sugere-se a exploração inicial de uma pequena sala que contenha, preferencialmente, uma colecção muito visitada. Deste modo, o investimento inicial não será demasiadamente elevado, podendo-se tirar o maior aproveitamento e efectuar os melhores testes com um bom número de visitas.

### 5.3 Análise de Requisitos

Na secção 4.2.1 foram apresentados requisitos genéricos aos quais a arquitectura deve responder, mas neste ponto acrescentam-se requisitos essenciais visando as particularidades do protótipo a ser implementado.

Em termos de funcionalidade, para este protótipo pretende-se que o requisito que considera o utilizador como elemento passivo seja fortemente seguido, fazendo com que o sistema ofereça uma interacção intuitiva, simples e o menos intrusiva possível. Quanto a requisitos funcionais há ainda a destacar os seguintes:

- É essencial que o visitante fique registado no sistema tornando-se um utilizador com um perfil associado, mesmo que seja o perfil genérico;
- O registo deve ser feito à entrada quando adquire o bilhete ou, caso já o tenha, numa área de entrada destinada para o efeito;
- O utilizador deve receber uma etiqueta para usar como *badge* ao pescoço;
- A(s) antena(s) devem ser colocada(s), o mais discretamente possível, junto a cada objecto

para não chocarem com o ambiente da sala;

- A distância a que um utilizador é detectado deve variar de peça para peça. Este ponto deve depender do tamanho e posição da peça, assim bem como da popularidade da mesma pois pode atrair mais ou menos visitantes em simultâneo. A disposição das peças na sala também tem influência na parametrização da distância de detecção para cada peça.
- A duração da apresentação de informação adicional dirigida para um utilizador deve ser inversamente proporcional ao número de utilizadores simultaneamente detectados na área. Também deve ser bem escolhida (filtrada) e o mais sucinta possível pois num ambiente destes é natural que por vezes se formem algumas aglomerações de pessoas junto à peça. Dependendo da peça, do tamanho do ecrã, e da precisão do RFID, a partir de um determinado número de utilizadores a informação a visualizar terá que ser apenas genérica.
- O número de utilizadores presentes na sala/exposição também deve ser tido em conta para a configuração (automática) dos tempos de apresentação da informação por peça.

Quanto a requisitos não funcionais, os maiores cuidados centram-se na escolha do equipamento RFID. Este deve ser o mais preciso possível uma vez que podem estar vários utilizadores junto a um objecto em simultâneo.

### 5.4 Instanciação da Arquitectura

Nesta secção descreve-se a implementação do sistema quanto à instanciação dos vários módulos centrais em termos de *hardware* e *software*. Explicam-se as opções tomadas tanto para os módulos que podem ser directamente implementados noutros cenários como para o componente de núcleo aplicacional, considerado mais específico para esta implementação.

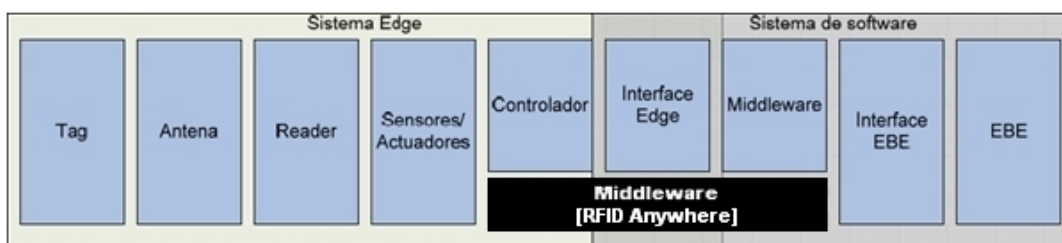


Figura 5.1: divingForPearls: Arquitectura típica de um sistema RFID [adaptado de [11]].

A Figura 5.1 apresenta uma arquitectura para sistemas que utilizam RFID, encaixando-se perfeitamente na arquitectura proposta pelo presente trabalho. Os primeiros quatro módulos do componente Sistema Edge correspondem ao *hardware*, sendo utilizados pela Colecção de Dispo-

sitivos da arquitectura do módulo de rastreio. O módulo Tag corresponde, mais exactamente, à etiqueta que identifica o utilizador, encontrando-se fora da camada do objecto, mas tendo que ser também conhecida pela colecção.

O controlador é usualmente integrado no próprio leitor, mas em termos lógicos integra, em conjunto com a Interface Edge e o *software middleware*, uma plataforma de *middleware* mais complexa que corresponde à segunda camada do componente de rastreio da arquitectura proposta.

Já a chamada lógica de negócio da aplicação é implementada pelos dois últimos módulos da arquitectura RFID. A interface EBE, que vai corresponder ao Gestor de Rastreio, traduz a saída do módulo Middleware para que a aplicação a possa entender. O módulo EBE corresponde à aplicação específica do sistema RFID, integrando todos os outros módulos principais da arquitectura da Figura 4.3.

#### 5.4.1 Componente de Rastreio

Este componente pode ser facilmente reutilizado noutros cenários pois não é específico desta aplicação. A configuração física do *hardware* é feita para o ambiente de aplicação e teria que ser adaptada, mas todos os módulos de interfaces de programação estão desenvolvidos de modo a poderem ser utilizados por outras aplicações.

Nesta secção é importante descrever, acima de tudo, o *hardware* utilizado na implementação e o pacote utilizado na camada de *middleware*. As escolhas efectuadas quanto a estes pontos foram determinantes na adopção da tecnologia de programação para o componente aplicacional. As opções tomadas são descritas nas próximas subsecções.

##### 5.4.1.1 Dispositivo de Rastreio

Hoje em dia, são muitas as opções que existem no que toca à escolha do *hardware* RFID. São muitos os produtos oferecidos por empresas como a Alien<sup>3</sup>, a OMRON<sup>4</sup>, a Intelleflex<sup>5</sup>, a ThingMagic<sup>6</sup>, ou a Zebra<sup>7</sup>, só para citar algumas das maiores referências deste mercado. Uma vez que a escolha das *tags* não oferece grandes dificuldades, optou-se por dedicar especial atenção à escolha do leitor e respectivas antenas. A escolha deste item também estará dependente da

---

<sup>3</sup><http://www.alientechnology.com/>

<sup>4</sup><http://www.omronrfid.com/>

<sup>5</sup><http://www.intelleflex.com/>

<sup>6</sup><http://www.thingmagic.com/>

<sup>7</sup><http://www.zebra.com/>

#### 5.4. INSTANCIACÃO DA ARQUITECTURA

---

solução de *middleware* adoptada. Para se ganhar em tempo de desenvolvimento, pretendia-se usar uma plataforma já existente que facilitasse a integração do *hardware* com a programação da aplicação. Deste modo, a escolha do *hardware* foi feita em paralelo com a escolha do *middleware*. Um requisito essencial seria o *middleware* ter conectores para o leitor escolhido. Um outro passaria pela possibilidade de realizar testes antes de se efectuar a aquisição do *hardware*. Pretendia-se utilizar um conjunto que desse garantias de qualidade e que se assemelhasse, ou fosse, aquele que seria utilizado no cenário real de aplicação, e nestas condições os custos de um kit destes são consideráveis (acima dos 2.000,00 euros).

São vários os tipos de leitores existentes, variando de acordo com a(s) sua(s) frequência(s) de trabalho, o seu carácter de mobilidade, as interfaces de comunicação ou o *firmware* que apresenta. A maioria dos leitores lê em todas as frequências usadas pelas *tags*, ou em espectros específicos de determinado tipo de *tag*, existindo também os que permitem uma configuração destes parâmetros (característica do *firmware*). Além do preço, a questão da mobilidade torna-se uma característica fundamental para a escolha do leitor. Para o presente protótipo, a escolha recaiu sobre um leitor fixo (ver Figura 5.2, imagem da esquerda) uma vez que o respectivo uso é feito na camada do objecto.

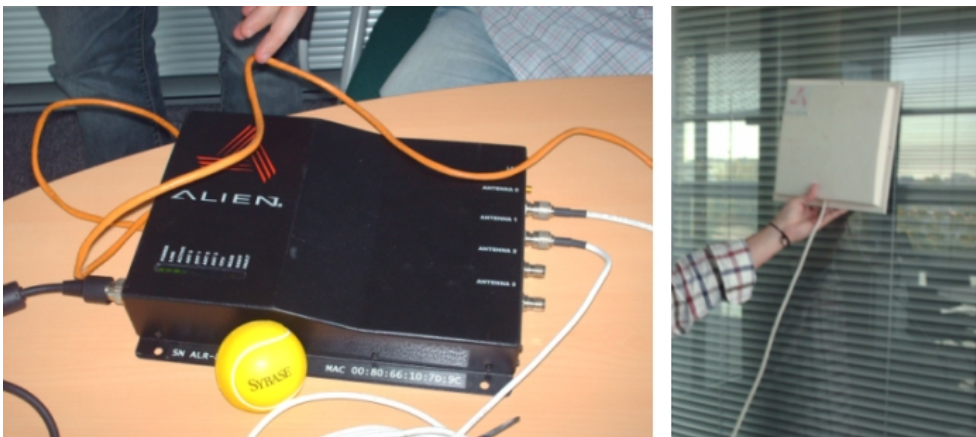


Figura 5.2: divingForPearls - Dispositivo de Rastreo: (esquerda) Leitor Alien e (direita) antena RFID utilizada.

Assim sendo, foi utilizado um kit da Alien, o ALR-8800 DEVKIT, para o qual a solução *middleware* escolhida tinha conectores. Uma vantagem oferecida por este conjunto foi o facto de existir no centro RSC da Sybase, podendo assim ser utilizado livremente para experimentação no protótipo. O kit da Alien é composto por um leitor modelo ALR-8800, duas antenas modelo ALR-8610 e 150 *tags* EPC Squiggle Gen 2. O leitor ALR-8800 consiste numa solução de

Frequência	865.6 MHz - 867.6 MHz
Protocolos <i>Tag</i> suportados	EPC Classe 1 Gen 2, EPC Classe 1 Gen1, ISO 18000-6c
Protocolos de rede	DHCP, TCP/IP, SNTP, DNS, SNMPv.3
Antenas	2 antenas de polarização circular
Portas antenas	4 portas para 4 pontos de leitura
Comunicação	RS-232, LAN TCP/IP
<i>Software</i> de suporte	Java e APIs .NET e demos
Canais	10

Tabela 5.1: divingForPearls - Dispositivo de Rastreo: Principais características do leitor RFID ALR-8800 da Alien.

elevado desempenho, susceptível de actualização para futuros protocolos, permitindo uma fácil integração e implementação de soluções RFID. Este dispositivo é suportado por kits de desenvolvimento de *software royalty-free* para Java e .NET. Para possibilitar a interacção com outros dispositivos, como um servidor, os leitores são fabricados com várias portas de comunicação, incluindo RS-232 e LAN TCP/IP, no caso do ALR-8800.



Figura 5.3: divingForPearls - Dispositivo de Rastreo: (esquerda) *Badge* com *tag* adesiva e (direita) exemplo de *tag* utilizada.

O kit da Alien possui *tags* passivas EPC Classe 1 Gen2 de elevada fiabilidade, eficiência e desempenho, com um custo reduzido. Podem ser aplicadas nas mais diversas situações, possuindo uma capacidade de 128 bits (96 bits para dados de utilização). São *tags* do tipo adesivo e no protótipo foram testadas em *badges* ao pescoço (ver Figura 5.3, imagem da esquerda). Além das *tags* contidas no kit, foram ainda utilizadas outras *tags* passivas idênticas (ver Figura 5.3, ima-

gem da direita), mas fabricadas por outras empresas (e.g., Kennedy Group e Avery Dennison). Optou-se por *tags* passivas porque se pretendia detectar os utilizadores apenas nas imediações dos objectos, tendo sido feita uma parametrização no leitor para que as *tags* fossem detectadas a distâncias entre 1 e 3 metros, dependendo do objecto em questão. Para os testes foram utilizadas quatro antenas ALR-8610, correspondendo cada uma a um determinado objecto. O leitor é identificado pelo servidor por TCP/IP na rede LAN.

### 5.4.1.2 Middleware

Esta camada é de extrema importância para o correcto funcionamento da infra-estrutura, sendo a responsável pela ligação da camada de *hardware* de baixo nível com a camada superior que é a interface directa com o componente aplicacional.

Observando novamente a Figura 5.1, verifica-se que o Controlador funciona como agente intermediário que possibilita que uma outra entidade comunique com o leitor. A Interface Edge já integra a aplicação com o *hardware* RFID, tendo como principal função obter informação do leitor e utilizá-la para activar as associações aos actuadores e anunciadores no exterior. Quanto ao módulo Middleware, este integra tudo o que se encontra entre a Interface Edge e a lógica aplicacional do sistema, fazendo a abstracção da ligação com o *hardware* para a aplicação.

Adaptando a arquitectura proposta a um sistema RFID, a melhor solução para o presente módulo passa pela adopção de uma plataforma *middleware* existente que considere o Controlador e a Interface Edge. A alternativa seria a utilização das interfaces de programação (APIs) disponibilizadas pelo fabricante do *hardware* para se fazer o desenvolvimento da interface *middleware* da arquitectura. Neste caso, teria como desvantagem a dependência do fabricante, obrigando à programação de vários módulos do tipo conector por dispositivo físico. A vantagem óbvia seria a obtenção do *software* sem custos adicionais, além de ficar à medida das necessidades. Contudo, para o presente protótipo pretendia-se implementar uma solução rápida que permitisse testar a arquitectura e o conceito da aplicação, não sendo relevante o desenvolvimento do *middleware*. Pretendia-se ainda adoptar uma solução robusta que permitisse várias parametrizações para um sistema complexo e diversificado, como o apresentado.

Para o presente protótipo, a escolha centrou-se entre uma proposta da Sybase/iAnywhere, o RFID Anywhere [147], e outra da Sun Microsystems, o Sun Java System RFID [148]. Ambos os pacotes são fáceis de obter, de instalar e oferecem um acesso gratuito, podendo ser livremente utilizados para fins académicos e de experimentação. As versões testadas foram as mais recentes, o que corresponde à terceira em ambos os casos, e tanto um como o outro oferecem

um ambiente de simulação sem recurso a *hardware*. Baseiam-se nas normas mais comuns da indústria, incluindo os definidos pela EPCglobal, e apresentam arquitecturas de componentes similares (ver Figura 5.4). Ambas as plataformas fornecem níveis elevados de confiabilidade e escalabilidade numa rede EPC, simplificando a tarefa de integração com múltiplos sistemas e dispositivos existentes. Apresentam conectores para um largo conjunto de leitores dos mais diversos fabricantes e conseguem processar e filtrar os dados, fornecendo apenas informação útil sobre identidade e eventos, e resguardando tanto a rede como as aplicações de uma sobrecarga de informação. Ambos os pacotes suportam especificações *EPCglobal Application Level Events* (ALE) para um processamento e uma filtragem de dados inteligentes. Quanto a plataformas onde podem ser utilizados, o RFID Anywhere encontra-se limitado à Windows, enquanto que a Sun tem soluções para SPARC, Linux e Windows. A proposta da Sun parece seguir um caminho mais normalizado e com maior poder de migração, mas verificou-se que, actualmente, o RFID Anywhere é mais simples, mais consistente e com mais e melhor informação disponível.

Após alguns testes, dada a similaridade encontrada nos pontos centrais, a escolha recaiu na proposta da Sybase por ter sido mais fácil trabalhar com a mesma. Consideram-se ainda como pontos determinantes para a escolha a facilidade de montagem de uma simulação e de integração do leitor adoptado, assim como o facto de existir apoio por parte do laboratório Sybase RSC, o que contribuiu para uma implementação e uma experimentação mais rápida deste módulo da arquitectura.

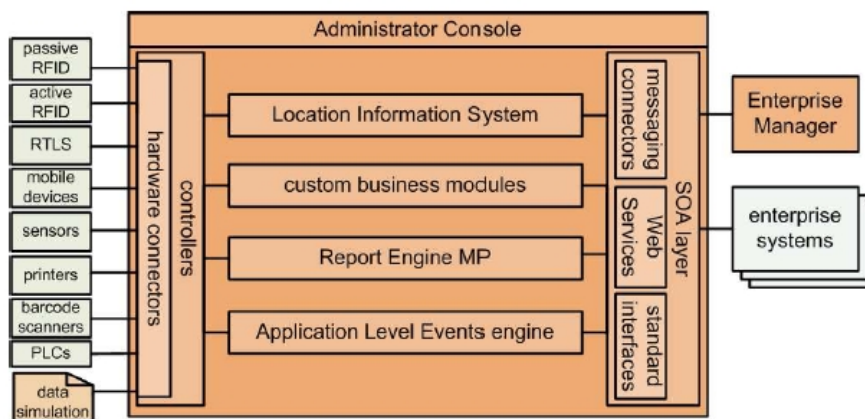


Figura 5.4: divingForPearls - RFID Anywhere: Relações entre os seus Componentes.

A plataforma RFID Anywhere integra vários componentes (ver Figura 5.4), de onde se destacam os conectores, os controladores, o simulador, os módulos centrais para filtragem de dados e lógica de aplicação, a consola de administração e a extensão para o Microsoft Visual

Studio. No apêndice A pode ser encontrada uma descrição detalhada dos vários componentes.

Após a escolha da plataforma *middleware*, a configuração do presente módulo passou pelas seguintes fases:

1. Uma vez que o RFID Anywhere apenas pode ser instalado numa máquina com Windows XP Professional (com SP2) ou Windows server 2003, escolheu-se a segunda opção para o protótipo *divingForPearls*, tendo sido necessário instalar componentes Microsoft adicionais: o MSMQ Messaging Service, o servidor Web IIS (*Internet Information Services*), a plataforma .NET 1.1 SP1, e a extensão de *web services* (WSE) 2.0 SP3. No sistema, esta máquina funciona como servidor RFID que deve conhecer o(s) leitor(es) RFID que se encontra(m) ligado(s) à rede.
2. Posteriormente, foram desenvolvidos alguns testes através do componente de simulação (detalhes no Apêndice A). Estes visaram a parametrização do sistema e a programação do Gestor de Rastreo e dos módulos da aplicação, sem que para tal fosse necessário utilizar o *hardware* numa primeira fase.
3. Finalmente, foi criado o *Business Module* (BM) *divingForPearls* para ser utilizado pelo módulo Gestor de Rastreo, visando a disponibilização dos dados para o módulo Interpretador Contextual do componente aplicacional. Após ter sido criado, foi adicionado através da consola de administração aos restantes conectores, sendo executado como serviço local (este e os serviços criados na simulação podem ser vistos na Figura A.1). Foi ainda criado um conector *messaging* TCP, associado à porta 10002, para ser utilizado pelo *Business Module* para disponibilização dos dados.

### 5.4.1.3 Gestor de Rastreo

A lógica da aplicação é implementada pelos dois últimos módulos da arquitectura RFID. A interface EBE (ver Figura 5.1), que corresponde ao Gestor de Rastreo, serve para efectuar a integração do módulo anterior com a aplicação propriamente dita. Esta interface é adaptada à solução de *middleware* utilizada. Sendo responsável pela abstracção da tecnologia de rastreo, o Gestor disponibiliza, para o módulo aplicacional, os dados das interacções de uma forma clara e perceptível para o último.

Assim sendo, utilizou-se o BM *divingForPearls* como Gestor de Rastreo, pois este envia os dados pretendidos em documento XML para o Interpretador Contextual do módulo Gestor de Informação (núcleo aplicacional). O próprio BM poderia mapear os dados em bruto das *tags* e antenas de acordo com um ficheiro XML secundário, mas apenas envia os dados da detecção

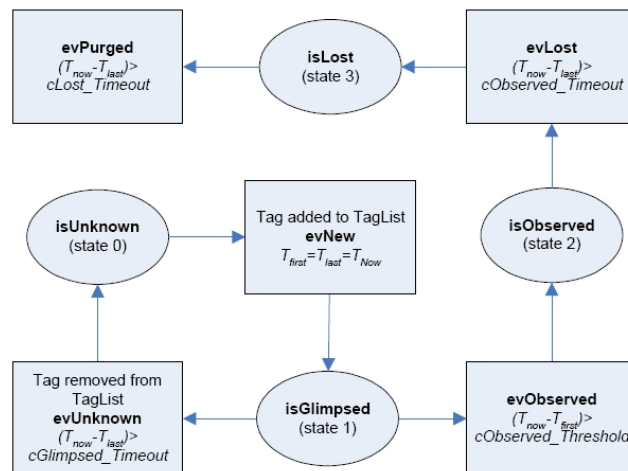


Figura 5.5: divingForPearls - Gestor de Rastreo (BM divingForPearls): Os 4 estados considerados pelo BM para a actividade de *tags*.

para serem interpretados pelo módulo central. O documento XML criado pelo BM é enviado por TCP sempre que um evento de observação ou de perda de uma *tag* ocorre (ver Figura 5.5). Para o presente cenário, o BM apenas ficou programado para disponibilizar em XML o instante (data e hora) em que uma *tag* é observada ou é perdida, além das respectivas identificações da *tag* e da antena/leitor que despoletou o evento (ver Figura 5.6).

```

<?xml version="1.0" encoding="UTF-8" ?>
- <Detection eventType="observed">
  <Source>LeitorGrp1\\Antena1</Source>
  <TagID>303800004000008000000005</TagID>
  <TagType>EPC_SGTIN96</TagType>
  <Date>2007-10-27</Date>
  <Time>13.30.05</Time>
</Detection>

```

Figura 5.6: divingForPearls - Gestor de Rastreo: XML de uma detecção criado pelo BM.

O BM também pode disponibilizar métodos via *web services* que permitem que seja questionado acerca dos dados. Este procedimento funciona como alternativa à solução de envio automático dos relatórios XML para o núcleo aplicacional.

#### 5.4.2 Componente Aplicacional

A componente aplicacional engloba os módulos Personalização e Conteúdos de Objectos, além do Gestor de Informação (ver Figura 4.3). Observando novamente a Figura 5.1, verifica-se que este componente corresponde ao módulo EBE na arquitectura RFID.

Para a presente aplicação, não foram desenvolvidos os módulos relativos ao Gestor de Pedidos e ao Cache pois neste cenário não existem dispositivos móveis que possam fazer pedidos ao sistema central. Por outro lado, também não foram desenvolvidos os módulos do Gestor de Eventos e da Fila de Eventos. A utilização de apenas um leitor com quatro antenas e a robustez do pacote *middleware* corresponderam a todas as expectativas, possibilitando uma parametrização detalhada em termos de eventos. A actividade em redor das antenas também foi relativamente baixa, tendo o BM *divingForPearls* efectuado uma gestão eficaz dos eventos recebidos a partir do *hardware*. Uma boa configuração, para fazer face a um maior número de antenas e *tags* nas suas imediações, poderia passar pelo desenvolvimento de BM's por cada grupo de quatro antenas, tendo também a atenção de configurar no mesmo grupo antenas de menor actividade com outras de maior. Com este processamento distribuído, o Gestor de Eventos poderia assim ficar atento às portas TCP dos vários BM's, organizando e passando os vários documentos XML para o Interpretador.

A linguagem de programação utilizada para o desenvolvimento dos módulos do componente aplicacional foi a **C#** (C-Sharp). Tanto os módulos Interpretador Contextual e Gestor de Composição, como as aplicações para gestão de clientes e objectos e para gestão de perfis, foram desenvolvidos com recurso a esta linguagem. Inicialmente, pensou-se em utilizar Java, mas com a escolha da plataforma de *middleware* e com a programação do BM *divingForPearls* (o Gestor de Rastreo) concluiu-se que, tanto por uma questão temporal como de integração, seria de aproveitar o facto de a plataforma de desenvolvimento já se encontrar preparada.

O componente aplicacional engloba três bases de dados, tendo sido escolhido como Servidor Gestor de Bases de Dados (SGBD) o **MySQL DB server** [149]. Este é um *software* Open Source de fácil utilização e de rápida instalação. Tem sido o servidor de bases de dados com maior crescimento de mercado, sendo fiável e apresentando uma boa relação entre potencialidades e recursos em termos de máquina. Sendo popular e muito utilizado na área das aplicações Web, apresenta-se com um bom desempenho quando solicitado por vários pedidos simultâneos, possuindo um conjunto de soluções menos complicadas do que os outros SGBD's que são as referências de mercado, o Oracle<sup>8</sup> e o Microsoft SQL Server<sup>9</sup>.

Tal como já foi abordado, o **XML** é utilizado para a formatação e uniformização da comunicação de dados entre o módulo de Rastreo e o módulo Gestor de Informação. Mesmo no interior deste último, os dados são trocados entre o Interpretador, o GdC e o Servidor em

---

<sup>8</sup><http://www.oracle.com/>

<sup>9</sup><http://www.microsoft.com/sqlserver/>

<b>Perfil</b>	<b>Campos de Preferência</b>
0	Genérico - sem preferências
1	Arte, Política e Desporto
2	Política, Literatura e Arte
3	Desporto, Arte e Literatura
4	Literatura, Política e Desporto

Tabela 5.2: divingForPearls - Personalização: Perfis criados.

formato XML. Esta relevância surge do facto de o XML ser uma forma bastante flexível de representar informação, oferecendo grande portabilidade e reutilização do sistema e facilidade na sua evolução.

Nas próximas subsecções são descritas as opções tomadas para cada um dos módulos e sub-módulos deste componente.

#### 5.4.2.1 Módulo Personalização

Este módulo é composto por uma base de dados que armazena os dados privados dos utilizadores (visitantes de um museu) e os perfis e preferências criados para o cenário de aplicação. Nesta aplicação não existem dados de utilizador públicos, uma vez que se pretende apenas conhecer o utilizador para que a informação adicional sobre um objecto vá ao seu encontro. Alguns dados adicionais podem ser pedidos para uma correcta identificação em futuras actividades lúdicas e promocionais. O registo do visitante deve ser rápido, exigindo assim um reduzido conjunto de dados sobre o mesmo. Pela mesma razão, já existem perfis de acordo com um conjunto de gostos/preferências para que o visitante possa escolher aquele com que mais se identifica.

A Tabela 5.2 apresenta cinco perfis exemplificativos que foram utilizados nos testes feitos ao sistema. O primeiro existe para todos os visitantes que, querendo utilizar o sistema, não se identificam com um conjunto de preferências pré-definido.

Adicionalmente, foi criada uma pequena aplicação para gestão dos perfis, permitindo a adição de preferências e a criação e actualização de perfis de acordo com as áreas em que se inserem as peças em exposição.

#### 5.4.2.2 Módulo Conteúdos de Objectos

A segunda base de dados do sistema encontra-se neste módulo. Esta é exclusivamente dedicada aos conteúdos multimédia que compõem a informação adicional respeitante a um objecto.

Os conteúdos, sejam eles texto, imagem, áudio, ou vídeo, relacionam-se não apenas com o objecto, mas também com o perfil. Visto conter dados que podem ser mais pesados para o sistema, dependendo da especificidade do cenário de implementação, pode ser necessário ter uma máquina dedicada a este repositório.

Este módulo levanta algumas questões quanto ao melhor tipo de repositório a utilizar para dados multimédia, mas este é um tema que foge ao objectivo da presente dissertação. No protótipo divingForPearls foram utilizados os vários tipos de campos BLOB (*Binary Large Object*) para armazenar áudio e imagem em conjunto com o sistema de ficheiros, onde ficaram armazenados os vídeos. Uma outra solução que poderia ter sido implementada passaria pela utilização do SGBD da Oracle com a ferramenta Oracle Multimedia<sup>10</sup> (originalmente conhecida por InterMedia) para este módulo dos objectos. Pensa-se que, de futuro, esta possa ser a melhor solução pela fiabilidade, robustez e eficientes gestão e recuperação de conteúdos multimédia.

### 5.4.2.3 Interpretador Contextual

Este módulo, programado em C#, funciona como decisor do sistema ao processar a informação contextual recebida do módulo de rastreio para determinar que informação adicional deve ser mostrada e em que ecrã/objecto. É composto, na essência, por dois pacotes (sub-módulos) de classes determinantes.

O primeiro escuta a porta 10002 por onde recebe os novos documentos XML disponibilizados pelo BM divingForPearls via conector TCP. Ao receber um novo documento XML com indicação da detecção faz o processamento imediato, actualizando de seguida a respectiva base de dados. Esta é mapeada a partir do diagrama de classes da arquitectura do Interpretador, vista na Figura 4.4. De acordo com a informação recebida em XML (ver Figura 5.6), o processo de actualização passa pelo acesso à tabela Interação, com o seguinte procedimento:

- Se (atributo *eventType* == "observed") - inserção de novo registo utilizando para tal os valores das marcas Source, TagID, e Date/ Time (como instante inicial de interacção);
- Caso (*eventType* == "lost") - utilização dos valores das marcas Source, TagID e Date para validar a detecção contra um registo de interacção existente que não tenha o atributo do instante final preenchido, e actualização do mesmo com o valor de Time.

Um outro pacote de classes questiona a base de dados de acordo com o Source, para saber que objecto foi "visitado", e com a TagID, para determinar o "visitante". Essencialmente, há

---

<sup>10</sup><http://www.oracle.com/technology/products/intermedia/index.html>

que descobrir o perfil do utilizador possuidor da etiqueta com o identificador TagID. Sabendo o objecto junto ao qual se encontra o utilizador, determina-se que informação adicional deve ser mostrada. A partir destes dados é construído um novo documento XML que é passado ao módulo Gestor de Composição. A Figura 5.7 contém dois exemplos de documentos XML que darão as indicações ao Gestor de Composição para a preparação da visualização da informação. No documento da esquerda, verifica-se que são dois os utilizadores junto ao objecto, aparecendo o que tem a interacção mais recente em primeiro lugar. O outro documento é uma actualização do primeiro, mostrando que apareceu um terceiro utilizador, mas, entretanto, o segundo desapareceu da zona. O primeiro utilizador mantém-se junto ao objecto desde o instante inicial.

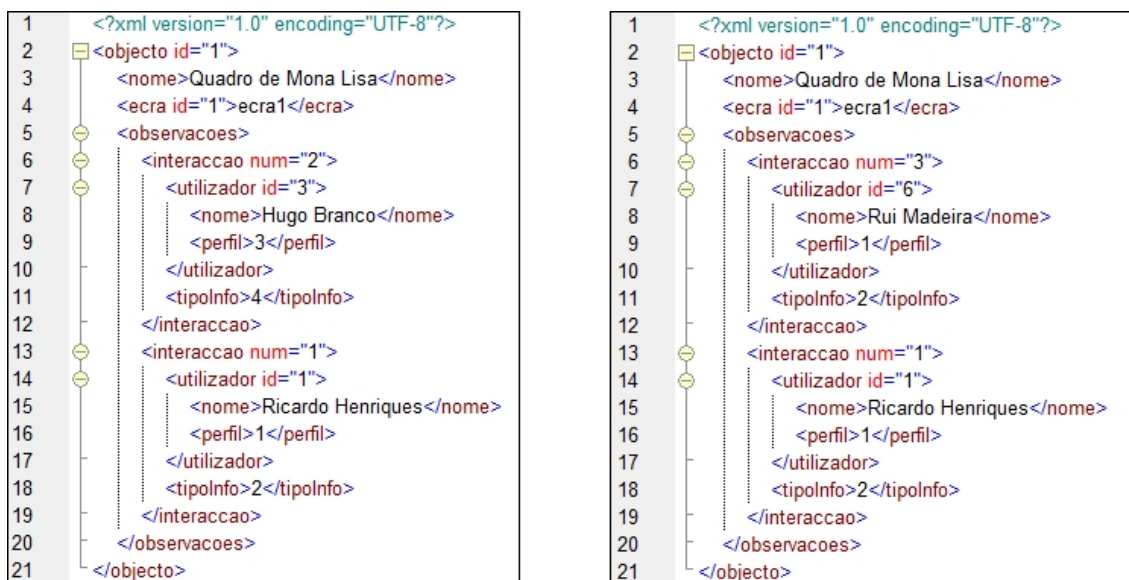


Figura 5.7: divingForPearls - Interpretador Contextual: Documentos XML passados para o Gestor de Composição.

#### 5.4.2.4 Gestor de Composição

Este módulo é responsável pela colocação de um outro documento XML na directoria correcta do Servidor de Informação. Antes de o fazer, processa o documento XML passado pelo Interpretador. Questiona o módulo Personalização para retirar alguns dados sobre o utilizador (é dependente da interface de visualização pretendida, mas na essência será o nome) e o módulo Conteúdos de Objectos para retirar a porção de informação que seja texto. O TipoInfo indicará se a informação adicional tem na sua composição texto, além do resumo genérico que já está incluído. Conteúdos como imagem ou vídeo serão retirados da base de dados apenas pelo submódulo Composição de Apresentação presente no componente Interface Ecrã. Esta distribuição

## 5.4. INSTANCIACÃO DA ARQUITECTURA

de processamento permite obter respostas mais rápidas, sendo vital para este tipo de sistemas, que exige rapidez de resposta. Também permite que o Servidor de Informação fique em posse de um documento que já contém informação adicional em forma de texto, para o caso de virem a existir dispositivos móveis fazendo pedidos ao Servidor.

O novo documento XML (ver Figura 5.8) criado por este módulo será colocado numa determinada directoria que se encontra no Servidor de Informação. O Gestor sabe qual é a directoria pela marca ecrã contida no documento recebido do Interpretador (ver Figura 5.7).

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <ecra id="1">
3   <objecto id="1">
4     <nome>Quadro de Mona Lisa</nome>
5     <info_generica>Mona Lisa é a mais notável e conhecida obra do pintor italiano Leonardo da Vinci.
6     O quadro apresenta uma mulher com uma expressão introspectiva e um pouco tímida. O seu
7     sorriso restrito, que deixa ver ligeiramente os dentes, é muito sedutor, mesmo que um pouco conservador.
8     </info_generica>
9   </objecto>
10  <utilizadores>
11    <utilizador ordem="1">
12      <nome>Ricardo Navarro</nome>
13      <perfil>3</perfil>
14      <tipolInfo>1</tipolInfo>
15    </utilizador>
16    <utilizador ordem="2">
17      <nome>Rui Madeira</nome>
18      <perfil>1</perfil>
19      <tipolInfo>2</tipolInfo>
20    </utilizador>
21  </utilizadores>
22 </ecra>
```

Figura 5.8: divingForPearls - Gestor de Composição: Um documento XML para o ecrã 1.

Finalmente, o Gestor faz um recarregamento (*refresh*) automático de um ficheiro XHTML que se encontra na mesma directoria onde colocou o documento XML.

### 5.4.2.5 Servidor de Informação

Para a presente aplicação da infra-estrutura, o Servidor de Informação é um servidor Web que recebe pedidos HTTP. A utilização deste servidor garante a portabilidade entre os vários tipos de dispositivos mesmo que estes sejam limitados.

No servidor foi criada uma directoria por ecrã/objecto. Os nomes atribuídos às directorias são iguais às designações atribuídas aos ecrãs correspondentes. É desta forma que o GdC sabe em que directoria colocar o documento XML gerado. Desta forma, um ficheiro XML chamado, por exemplo, "ecra1.xml" é colocado na directoria onde já se encontram outros dois ficheiros para o ecrã 1. Estes são um ficheiro XHTML e um Shockwave Director, que é a base da apresentação final para a informação adicional que visa aumentar o objecto (ou peça). A título de exemplo,

para a implementação testada foram criadas quatro directorias, originando um endereço HTTP de apresentação da informação adicional para cada um dos quatro ternos *objecto/ecrã/antena*.

Caso se pretenda, este servidor permite ainda que dispositivos móveis possam fazer pedidos dos ficheiros XML ou de páginas e apresentações criadas especificamente para os mesmos.

### 5.4.3 Componente Interface Ecrã

Fisicamente, este componente é constituído apenas pelas plataformas dos dispositivos de visualização. Embora o módulo Composição de Apresentação apareça incluído no módulo Interface Ecrã, esta é uma inclusão lógica pois em termos físicos este sub-módulo é incorporado no servidor Web descrito na subsecção anterior.

Em termos das tecnologias utilizadas para a apresentação da informação, optou-se por se ter uma base feita em XHTML (*eXtensible Hypertext Markup Language*), que é uma preparação para futuras visualizações em dispositivos móveis, e uma interface gráfica em ShockWave Director.

O XHTML<sup>11</sup> é uma reformulação da linguagem de marcação HTML, sendo baseada em XML. Combina os elementos de marcação HTML com as regras rígidas de XML. Este processo de padronização visa a exibição de páginas Web em diversos dispositivos (televisão, dispositivos móveis, etc), tendo a clara intenção de melhorar a acessibilidade. Não existem muitas diferenças entre o HTML e o XHTML. À medida que mais e mais ferramentas XML são disponibilizadas, tais como o XSLT (*EXtensible Stylesheet Language Transformations*) para transformação de documentos, maiores vão sendo as vantagens do XHTML, pois este integra-se na perfeição com a tecnologia XML. Em termos de apresentações, um documento XHTML pode incluir, por exemplo, SMIL (*Synchronized Multimedia Integration Language*) ou SVG (*Scalable Vector Graphics*), podendo trazer vantagens futuras para um sistema como o apresentado neste capítulo.

Uma vez que existia a ideia de se criar uma apresentação rica em termos multimédia, optou-se pela utilização da ferramenta **Adobe Director**<sup>12</sup>. Para além das boas capacidades de tratamento dos vários objectos multimédia, também possui mecanismos, através de programação, para a criação de conteúdos de maior complexidade. Permite a programação, importação, criação, interligação, sincronização, animação, publicação e exportação de conteúdos multimédia (imagem, som, vídeo digital, texto) complexas. O Director tem uma linguagem de programação incorporada e exclusiva, o Lingo, que apresenta bastantes semelhanças com outras linguagens, como JavaScript, C ou Java.

---

<sup>11</sup><http://www.w3.org/MarkUp/>

<sup>12</sup><http://www.adobe.com/products/director/>

De seguida passam-se a descrever as opções tomadas para cada um dos módulos.

### 5.4.3.1 Dispositivo de Visualização

Relativamente aos dispositivos utilizados no projecto divingForPearls, estes foram vulgares monitores TFT/LCD, tendo sido usado um por cada um dos quatro objectos da experimentação. Assim sendo, não foi necessário realizar trabalho adicional quanto à parametrização e gestão dos mesmos.

### 5.4.3.2 Gestor de Dispositivo

Para este sistema não foram criados e utilizados gestores específicos para os ecrãs utilizados. De todos os modos, uma vez que cada ecrã ficou com um computador dedicado, estes podem ser vistos como gestores. Através deles definiram-se as características gráficas de cada um dos ecrãs para visualização da informação, funcionando como clientes do Servidor de Informação. Para tal, cada um deles estava ininterruptamente ligado a um endereço específico no servidor, fazendo um simples pedido HTTP. Por exemplo, o computador atribuído ao ecrã 1, "divingPC1", tinha o *browser* ligado ao endereço:

- <http://divingServidorInfo/divingForPearls/ecra1/ecra1.html>

Desta forma, o computador designado por "divingPC1" era o responsável pela gestão da visualização da informação adicional no ecrã 1.

### 5.4.3.3 Apresentação e Composição de Apresentação

Um módulo Composição de Apresentação (CA) é composto por uma página XHTML e por um filme ShockWave Director. Para o sistema desenvolvido, a página Web serve essencialmente para integrar o filme para posterior visualização no *browser* do computador do ecrã cliente. Foi criada uma aplicação CA que foi replicada e ligeiramente adaptada, apenas em termos visuais, para cada objecto. Posteriormente, cada um dos CAs de ecrã (ficheiros XHTML e Shockwave) utilizados foi colocado na respectiva directoria no servidor Web de Informação.

Em termos de funcionamento, quando existe uma detecção junto a um objecto, o Gestor de Composição aplicacional cria e coloca o respectivo documento XML (ver Figura 5.8) na directoria correspondente do Servidor Web. Efectua o recarregamento automático da página Web, que recarrega o filme Shockwave. Este, ao reiniciar, lê o ficheiro XML dando início ao seu

ID	Descrição
0	sem informação para o perfil
1	texto
2	texto + imagem
3	imagem + vídeo
4	texto + imagem + áudio
5	imagem + áudio + vídeo

Tabela 5.3: divingForPearls - Composição de Apresentação: Valores definidos para TipoInfo.

processamento. Deste modo, o utilizador irá ver uma nova apresentação de informação adicional sobre o objecto no ecrã junto ao mesmo.

Primeiramente, o documento XML recebido teve que ser processado por um *parser* de forma a que os dados fossem retirados e estruturados. Torna-se importante referir este ponto porque o *parsing* de XML não é uma tarefa comum em filmes Director. São poucos os *parsers* existentes, com realce para duas soluções: o XML Parser Xtra, distribuído com o Director, ou o XML Flash Object. Estes pacotes de *software* são chamados de Xtras, e funcionam como bibliotecas externas, que visam dotar o Director de funcionalidades que este não possui. O primeiro é um Xtra do tipo *script*, enquanto que o segundo é original do Flash, tendo que ser importado para o Director como Xtra. Optou-se pela utilização do segundo por ser mais rápido e mais fiável, sendo este um factor determinante para o sistema. Sendo assim, foi criado um objecto do tipo *parent script*, responsável pelo *parsing* no interior do filme Director. Após o processamento da informação constante no XML, a apresentação reinicia já com o aparecimento de informação dirigida para o utilizador detectado junto ao objecto do ecrã.

Foram tidos alguns cuidados com a definição da interface de apresentação, como a preparação de um desenho diferenciado por tipo de informação a mostrar. Isto é, dependente de a informação ser, por exemplo, apenas texto, texto e vídeos, texto com imagens e áudio, etc, a apresentação faz uma adaptação automática mostrando diferentes interfaces. Para tal, foi usada a marca TipoInfo que aparece no XML, uma vez que é o seu valor que indica para que desenho da interface a apresentação deve saltar. A Tabela 5.3 indica que tipos de informação foram definidos para o sistema em função dos vários tipos de informação multimédia que se definiram para os objectos.

A Figura 5.9 (os quatro objectos presentes no teste simulavam peças de obras reconhecidas por uma questão de simplicidade) ilustra um tipo de apresentação que se pode ter, na qual aparece na zona superior uma imagem com um texto resumo genérico. Já na zona inferior

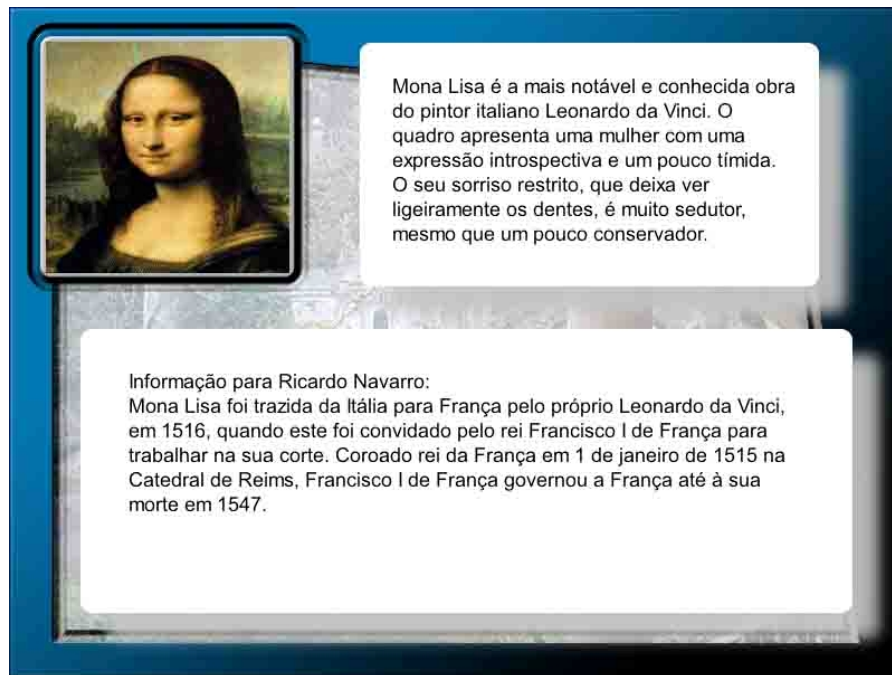


Figura 5.9: divingForPearls - Apresentação: Visualização de informação adicional sobre uma peça.

aparece a informação (apenas texto, referente a  $\text{TipoInfo} = 1$ ) dirigida para o tipo de perfil do utilizador que foi detectado na proximidade do objecto.

#### 5.4.4 Configuração Final

Nesta fase, sugere-se uma solução de configuração final da infra-estrutura para uma sala/zona do museu com oito peças. A configuração integra os componentes seguintes:

- 2 leitores ALR-8800;
- 8 antenas de polarização circular (4 por leitor);
- 1 servidor aplicacional (com *Middleware*, Interpretador Contextual, Gestor de Composição);
- 1 servidor Web (Servidor de Informação);
- 1 servidor de dados (Módulos de conteúdos e personalização);
- Dezenas ou centenas de *badges* com *tags* passivas;
- 8 ecrãs do tipo LCD ND4210w da ViewSonic (em alternativa, 8 TFT/LCD's vulgares ligados a 8 PC's);
- Ligações coaxiais entre antenas e leitores;
- Ligações LAN/WAN (TCP/IP) entre leitores, servidor aplicacional e os outros servidores.

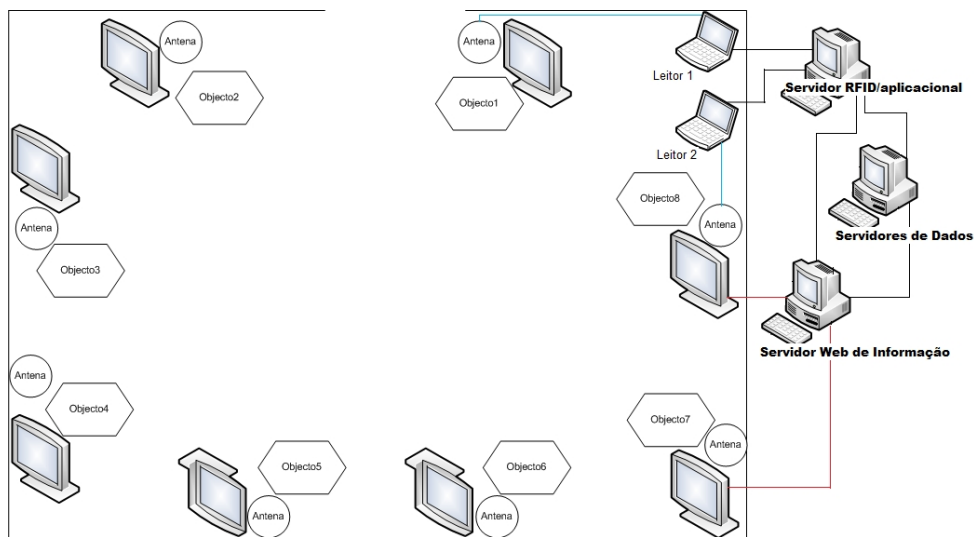


Figura 5.10: divingForPearls: Proposta de configuração lógica e física.

## 5.5 Aplicação para Gestão de Dados

O sistema inclui um módulo adicional para gestão dos seus elementos centrais, permitindo também a realização de simulações. Esta aplicação também foi desenvolvida em C#, interagindo com os módulos focados nos secções anteriores.

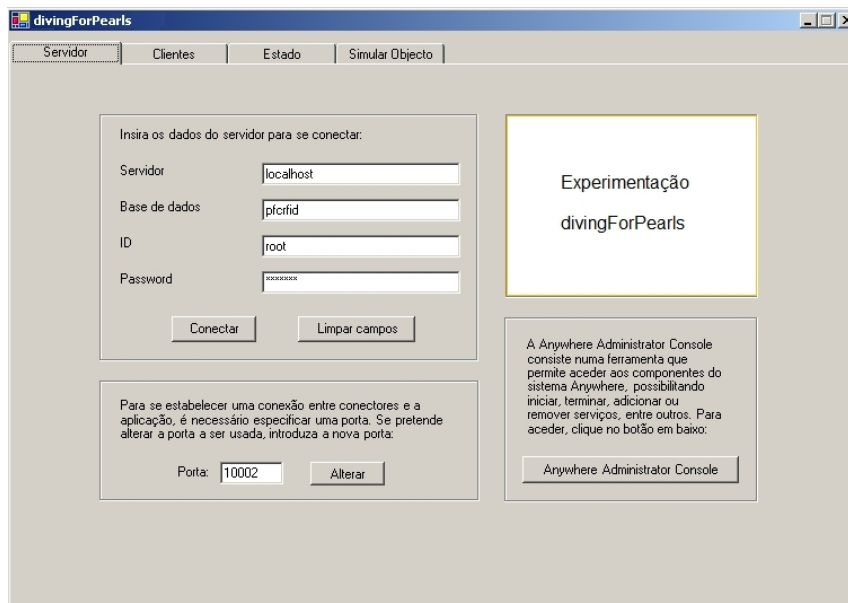


Figura 5.11: divingForPearls - Gestão de Dados: Ecrã "Servidor".

Ao ser iniciada, a aplicação começa por abrir a janela principal, que possui quatro ecrãs (ver

Figura 5.11): "Servidor", "Clientes", "Estado" e "Simular Objecto".

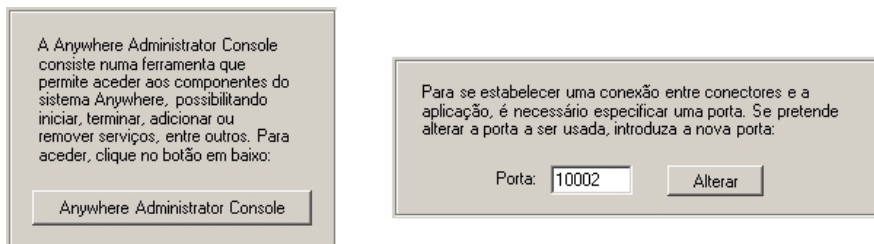


Figura 5.12: divingForPearls - Gestão de Dados: (esquerda) Ligação à consola de administração do RFID Anywhere e (direita) introdução da porta TCP para escolher BM.

O ecrã "Servidor" funciona como secção inicial onde devem ser introduzidos os dados para ligação ao(s) servidor(es) de dados onde se encontram as várias bases de dados. Para permitir o correcto funcionamento da aplicação, deve-se aceder à consola de administração do *middleware* RFID Anywhere para se iniciarem os serviços desejados. O *browser* abrir-se-á e será pedido o nome de utilizador e a palavra-chave, que são dados referentes à sessão do Windows. Ainda neste ecrã, pode ser alterada a porta de ligação por TCP ao *business module* pretendido. Por omissão, a aplicação irá receber os dados a partir do BM divingForPearls (ver Figura 5.12).

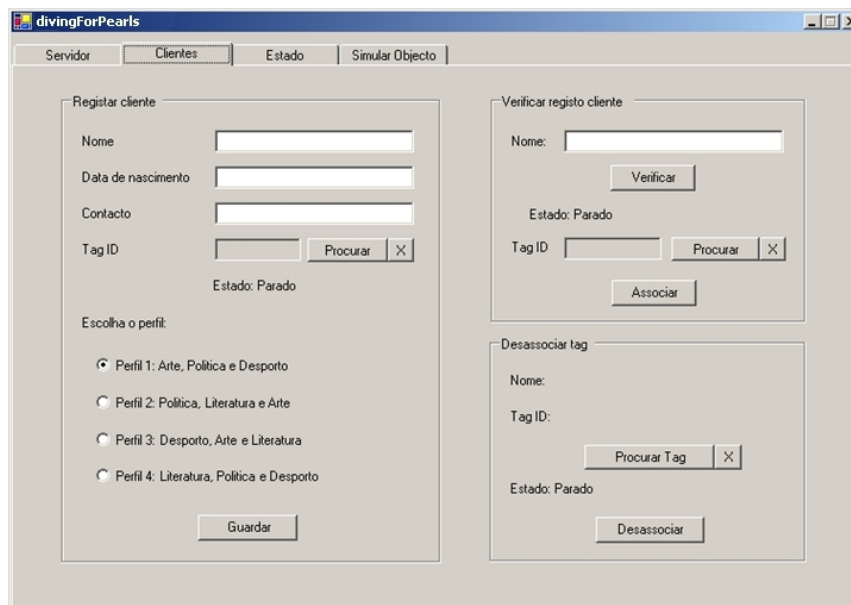


Figura 5.13: divingForPearls - Gestão de Dados: Ecrã "Clientes".

O ecrã "Clientes" (utilizadores) é utilizado para a inserção de novos utilizadores (ver Figura 5.13). A interface é simples e funcional para poder ser utilizada à entrada da sala/zona onde o sistema se encontra implementado. Um utilizador pode ser rapidamente registado, bastando

para tal que indique alguns dados e que escolha um perfil de entre aqueles que estão disponíveis. É-lhe então atribuída uma *tag* RFID que é passada num leitor de registo, associando-se automaticamente ao utilizador. Em termos de testes, também foram desenvolvidas as funcionalidades de associação de uma *tag* a um utilizador, caso este já esteja registado no sistema, e desassociação da *tag* ao utilizador, caso este deseje sair. Neste caso, a *tag* é passada no leitor, aparecendo os dados do utilizador para confirmação.

O módulo também permite que se saiba num determinado momento quem são os utilizadores que se encontram junto a um objecto. O ecrã "Estado" identifica os utilizadores activos junto ao objecto, quantos já entraram (apareceram) e quantos já saíram, apresentando também um historial. Esta secção tem a potencialidade de poder ser facilmente estendida para uma interface gráfica que vai indicando ao longo do tempo os utilizadores junto a cada objecto. Outras funcionalidades a serem adicionadas passariam pela determinação do percurso efectuado por um utilizador ao longo da sala e pela listagem dos objectos mais "visitados".

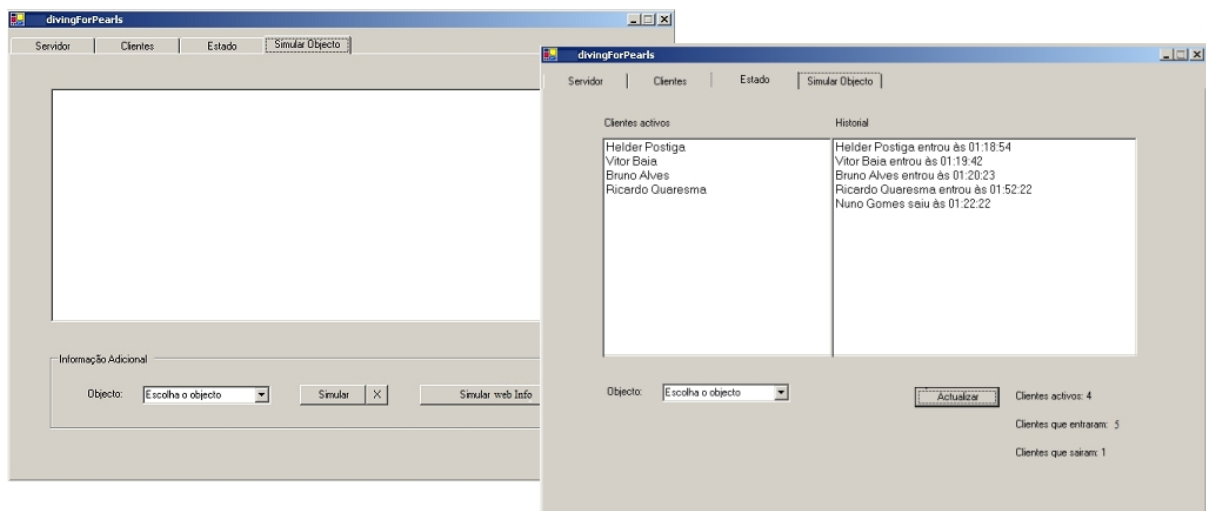


Figura 5.14: divingForPearls - Gestão de Dados: Ecrãs (esquerda) "Estado" e (direita) "Simular Objecto".

Por fim, o ecrã "Simular Objecto" foi desenvolvido com o propósito de se efectuarem simulações relacionadas com a apresentação da informação relativa a um objecto. Nela escolhe-se o objecto e a forma como se pretende visualizar a informação (como seria observada no ecrã correspondente). A opção por omissão apenas permite que seja mostrada informação em formato texto. Por outro lado, a opção "Simular Monitor" lança no *browser* a interface visual do módulo CA do objecto. Um administrador do sistema pode aceder em tempo real à informação adicional que vai passando num ecrã de um objecto da sala.



## Capítulo 6

# Realização do Protótipo directInfo2u

O projecto directInfo2u foi desenvolvido com o intuito de testar a aplicação da infra-estrutura num cenário do tipo local/instituição, visto como grande objecto. Num cenário destes, a aplicação poderia ser realizada num espaço privado, onde seriam conhecidos à priori todos os utilizadores diários do sistema, mas optou-se por efectuar a implementação numa zona pública, a sala de entrada comum de uma escola Yang's Martial Arts Association Portugal<sup>1</sup> (YMAAP), em Almada. Assim sendo, qualquer pessoa pode aceder a este local, tendo adicionalmente a oportunidade de beneficiar do sistema.

O sistema tem como tecnologia base para detecção da co-presença, e comunicação, a Bluetooth (BT), sendo programado em linguagem Java, nas plataformas J2SE (Java 2 Standard Edition) e J2ME (Java 2 Micro Edition). A utilização dos dispositivos móveis pessoais, como telemóveis e PDAs, é considerada, possibilitando a interacção activa por parte dos utilizadores. O sistema implementado foi apresentado na conferência UBICOMM2007<sup>2</sup> (International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies), dando origem a um artigo [25].

### 6.1 Tipo de Cenário e Actividades

Um edifício é um cenário natural para aumento com um sistema como o proposto por esta dissertação. Entende-se local como o espaço físico correspondente a uma instituição, por fazer parte da mesma, ou a um evento extraordinário, que também estará relacionado com uma instituição ou entidade. Imagine-se o *lobby* de um hotel, a zona de recepção de uma

---

<sup>1</sup><http://www.ymaaportugal.com/>

<sup>2</sup><http://www.iaria.org/conferences2007/UBICOMM07.html>

empresa/instituição, a entrada de um laboratório, ou um espaço onde decorra uma conferência. Em todos estes cenários pode-se pensar num único ecrã público situado de forma a receber os utilizadores do espaço ou colocado num ponto onde ocorra uma outra actividade central. Se no cenário anterior um ecrã correspondia a um objecto (peça) do museu, neste caso já se pode pensar que o ecrã corresponde ao próprio museu (um grande objecto), encontrando-se no seu interior, ou, até mesmo, numa parede do exterior.

Assim sendo, e considerando as dimensões de um objecto deste tipo, em determinadas implementações pode ser interessante abordar uma configuração que considere vários ecrãs. Estes podem ser dispostos ao longo de vários pontos num ambiente ou serem estrategicamente distribuídos por várias divisões no interior da instituição. Qualquer um dos ecrãs aumentará a entidade (objecto) com informação adicional, mas, dependendo do conceito a implementar, todos eles podem apresentar a mesma informação ou apresentarem informação relativa à actividade com a qual se relacionem. Esta configuração mostra que, na prática, a interacção se efectua entre o utilizador e o ecrã, e não com o objecto em si, apesar de o resultado ser aumentar o objecto ao qual o ecrã "pertence". Pode-se entender que o objecto segue o utilizador sentindo a sua co-presença ao longo dos vários espaços que o integram, interagindo com o utilizador através dos ecrãs.

A implementação num cenário deste tipo dependerá muito da instituição e das actividades que decorram na mesma. São muitas as possibilidades para utilização do sistema, mas na vertente menos intrusiva e menos exigente para o utilizador pode-se imaginar uma pessoa a entrar num edifício e um ecrã a responder à sua presença. Se assim for, o ecrã poderá apresentar informação genérica e curta, caso a pessoa não seja um utilizador registado. A informação até poderá variar e adaptar-se de acordo com o número de vezes que essa pessoa tenha sido detectada no passado, indicando que não se está na presença de um estranho total, apesar de não se saber quem é nem quais são as suas preferências. Se a pessoa não se encontra registada, não tendo recebido qualquer etiqueta que a identifique, e se o local for de acesso público, o dispositivo móvel pessoal poderá servir como etiqueta. Deste modo, pode-se aproveitar o facto de a maioria destes dispositivos integrar a tecnologia Bluetooth. À entrada do edifício poderá existir um aviso indicando que caso se deseje tirar algum partido do sistema implementado deve-se ter o Bluetooth activado no dispositivo. Neste segundo cenário, aproveita-se a ubiquidade dos dispositivos móveis, permitindo que utilizadores não registados possam interagir com um ecrã pró-activo.

O cenário mais interessante será aquele em que, mantendo-se o menos intrusivo possível e

não obrigando o utilizador a uma interacção activa, o objecto/ecrã reconhece quem está na sua presença. Para que tal aconteça, será necessário que o utilizador esteja registado, possuindo um determinado perfil assente nas suas preferências, actividades e necessidades. Este perfil será construído tendo por referência o objecto, actividades e espaços no objecto. Deste modo, quando o utilizador for detectado, a informação adicional terá um contexto baseado no utilizador, no objecto, e nas actividades e espaço onde ocorrem. Imagine-se um professor que ao entrar na recepção da instituição onde trabalha se depara com um ecrã que, ao reconhecê-lo, lhe transmite algumas notícias personalizadas relacionadas com a instituição. Pouco depois dirige-se para o gabinete onde tem o seu local de trabalho frequente, deparando-se à entrada com um outro ecrã que lhe dá informações relacionadas com as suas aulas e com projectos em que está envolvido. Aquando do registo, poderia ser atribuída uma etiqueta de identificação, em tecnologia RFID, e/ou anotar-se o endereço Bluetooth do dispositivo móvel pessoal do utilizador para funcionar como identificador. Este cenário junta os vários princípios defendidos por esta dissertação.

Por outro lado, o sistema poderá ser estendido para um nível em que o utilizador deixe de ter um papel passivo. Nesta perspectiva, o dispositivo móvel poderá ser, mais uma vez, utilizado na interacção entre o utilizador e o objecto/ecrã, através de uma aplicação instalada no seu dispositivo móvel no momento de registo. Deste modo, o objecto continuará a conhecer o utilizador, mas este também poderá ter um acesso directo ao objecto e ao(s) ecrã(s). Com uma aplicação cliente outros serviços poderão ficar à disposição do utilizador. Existirá ainda a vantagem de obter informações consideradas privadas sobre as suas actividades e que não seriam indicadas para uma apresentação num ecrã público. Numa situação deste tipo unem-se os dois conceitos para visualização de informação. Poder-se-á pensar ainda em alguns serviços adicionais, tais como: recepção de informação e armazenamento para posterior consulta e utilização; resposta a questionários; e troca de informações com o objecto, como por exemplo, para actualização dos dados do registo.

## 6.2 Enquadramento Organizacional

Mais uma vez, à semelhança do que aconteceu com o protótipo anterior, foi importante direccionar a implementação da infra-estrutura para um cenário real. Para tal, estabeleceu-se uma parceria com a YMAAP para o desenvolvimento de um projecto final de curso.

A YMAA<sup>3</sup> é uma instituição de renome internacional na área do ensino das artes marciais.

---

<sup>3</sup><http://www.ymaa.com/>

Foi criada pelo Dr. Yang, Jwing-Ming, em 1982 nos EUA, e desde então cresceu bastante, contando hoje com mais de 50 escolas espalhadas pelos vários continentes, entre as quais uma escola da YMAAP situada em Almada.

A implementação do sistema nas instalações da escola de Almada pode possibilitar que os seus utentes se encontrem facilmente actualizados sobre os temas relacionados com a mesma. Disponibiliza ao utente um meio com o objectivo de facilitar o acesso automático, personalizado e interactivo à informação sempre que se visita o local. Um indivíduo, ao chegar às instalações, pode ser automaticamente "envolvido" em informação direccionada para si, e de acordo com o seu perfil, não necessitando de a procurar num *website*.

Assim sendo, uma deslocação quotidiana às instalações da YMAAP oferece uma experiência única numa instituição deste género, sem obrigar a custos adicionais e aproveitando a utilização de dispositivos vulgares. Introduzindo apenas um computador servidor e um ecrã no ambiente, o sistema pode ser visto como uma espécie de *placard* digital dinâmico, que também pode ser interactivo. Numa configuração inicial, os custos são reduzidos face aos benefícios que a imagem de uma instituição como a YMAAP pode alcançar junto dos utentes, sobretudo dos mais jovens.

### 6.3 Análise de Requisitos

O funcionamento do sistema deve assentar em dois requisitos essenciais. Um deles é a pro-actividade do ecrã público partilhado que aumenta a instituição, enquanto que o outro é a possibilidade de interacção activa por parte dos utilizadores. Adicionalmente, destacam-se os seguintes requisitos funcionais:

- Deve ser detectada a co-presença dos dispositivos móveis dos utentes do espaço a alguns metros de distância, e um grande ecrã situado numa parede de zona de passagem.
- O registo de utilizadores deve ser feito junto ao(s) administrador(es) do sistema;
- Um utilizador do sistema, para ser considerado como tal, deve utilizar um dispositivo móvel "activado" (com tecnologia de detecção activada e registado);
- O utilizador deve possuir um perfil implícito traçado de acordo com o seu histórico na YMAA, isto é, de acordo com artes praticadas, eventos participados, aulas assistidas, etc;
- Um visitante do local pode ser detectado, tendo um perfil genérico, mas não pode ser registado como utilizador caso não esteja associado à YMAA;
- A informação disponibilizada deve visar, essencialmente, os seguintes tópicos: notícias gerais sobre a YMAA; novos artigos; aulas e instrutores; novos eventos; informações e

questionários pessoais;

- A duração da apresentação de um pedaço de informação adicional dirigida para um utilizador deve ser inversamente proporcional ao número de utilizadores simultaneamente detectados no espaço. A partir de um determinado número de utilizadores a informação a visualizar terá que ser apenas genérica;
- Para que a comunicação directa entre servidor e dispositivos clientes ocorra será necessário que os últimos façam a autenticação através da aplicação cliente instalada;
- As comunicações entre servidor e clientes podem ter como objectivo: o envio da informação adicional (que pode ser mais extensa do que aquela que aparece no ecrã) para o dispositivo pessoal do utilizador, para uma leitura posterior; o envio de informações privadas para o utilizador; o envio de questionários ao utilizador; pedido de informação por parte do utilizador; envio de respostas a questionários para o servidor;
- O sistema deve considerar 4 tipos de utentes para a aumentação da instituição: 1) Não registado (não é utilizador); 2) Registado (utilizador) sem aplicação cliente; 3) Registado com aplicação cliente.

Como requisitos adicionais há a destacar que os utilizadores devem usar dispositivos móveis com tecnologia BT activa para serem detectados. Preferencialmente, os dispositivos devem implementar uma interface Java (JSR-82) que permita a comunicação com o servidor por parte de uma aplicação móvel. Por outro lado, o computador servidor central deverá possuir comunicação BT, seja embutida ou através de adaptador USB.

## 6.4 Instanciação da Arquitectura

À semelhança do que aconteceu com o primeiro protótipo, nesta secção descreve-se a implementação da infra-estrutura no ambiente. Explicam-se as opções tomadas e as tecnologias usadas na instanciação da arquitectura proposta.

Após a escolha da tecnologia BT para detecção (e comunicação a curta distância), optou-se pela tecnologia Java para o desenvolvimento dos módulos de programação. Esta opção foi, essencialmente, determinada pelas bibliotecas para a implementação de BT e pela plataforma J2ME que o Java oferece.

A Figura 6.1 apresenta uma arquitectura típica para sistemas desenvolvidos em plataforma Java e que utilizam tecnologia BT. A arquitectura apresenta três camadas principais, sendo a superior dividida em outras duas.

## 6.4. INSTANCIACÃO DA ARQUITECTURA

As duas camadas inferiores implementam os protocolos e as camadas da típica arquitectura BT presente na Figura 3.2 (ver Subsecção 3.2.2.2). O BT inclui esta pilha (*stack*) de protocolos, sendo parte dela implementada em *software* e outra parte via *firmware*. A última corresponde à camada inferior que controla directamente o *hardware*, encontrando-se dependente dele. Já a camada seguinte implementa por *software* os restantes protocolos (e.g., RFCOMM e SDP) do *stack* e é uma implementação dependente do sistema operativo. A camada superior corresponde à aplicação específica que se quer desenvolver. A penúltima camada (neste caso, designada por "BlueCove") representa as APIs (*Application Programming Interfaces*) que permitem que as aplicações Java acessem aos protocolos BT.

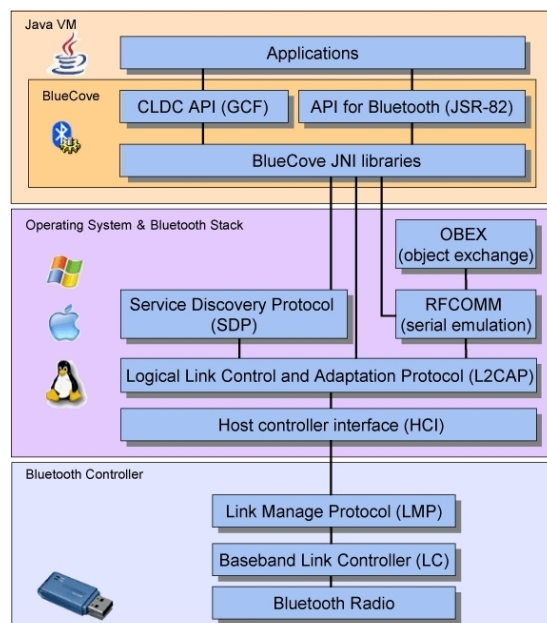


Figura 6.1: directInfo2u: Arquitectura detalhada de uma aplicação Java para Bluetooth [12].

### 6.4.1 Componente de Rastreo

Mais uma vez, a implementação deste componente é orientada para a reutilização noutros cenários. Os módulos Middleware e Gestor deste protótipo podem ser facilmente reutilizados noutras aplicações, mas o *hardware* também é de fácil adaptação, uma vez que é reduzido e de simples instalação.

A tecnologia BT foi escolhida por apresentar características muito positivas, como a velocidade, a fiabilidade e a capacidade de atravessar obstáculos, além de estar presente numa grande maioria dos dispositivos móveis pessoais carregados diariamente pelas pessoas.

A Figura 6.2 mostra os componentes da arquitectura BT/Java efectivamente seguida, observando-

se a separação lógica dos mesmos de acordo com os módulos do componente de rastreio.

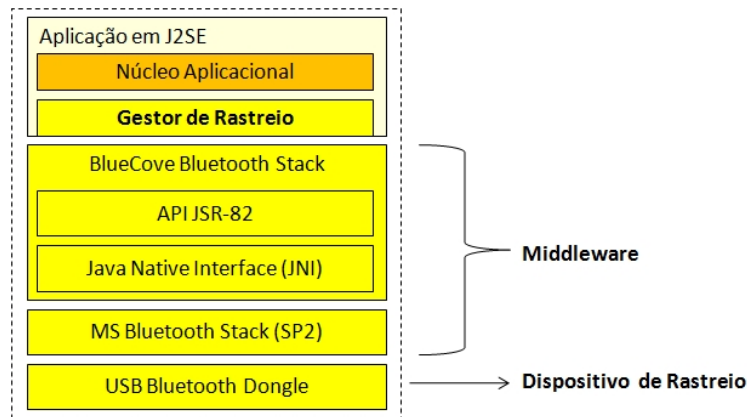


Figura 6.2: directInfo2u: Arquitectura Bluetooth/Java implementada para o servidor.

#### 6.4.1.1 Dispositivo de Rastreio

Nesta aplicação, o presente módulo é conhecido por dispositivo de BT local ("LocalDevice"), sendo um computador (o servidor BT), com capacidades BT. Estas podem ser devidas a um *chip* BT integrado de origem na máquina (algo que já acontece com qualquer computador portátil) ou a um adaptador de BT externo, sendo os de tipo USB muito comuns. Quando o adaptador USB de Bluetooth é ligado ao computador, o sistema operativo deve efectuar a gestão do adaptador através dos seus próprios *drivers*, não instalando controladores adicionais. Este procedimento é essencial para uma correcta e rápida configuração do sistema, sendo determinante para o funcionamento da camada Middleware. Deste modo, é necessário ter-se em atenção o tipo de adaptador que se utiliza, pois existem adaptadores que só funcionam com os seus *drivers*.

Para este projecto foram utilizados:

- um computador com Windows XP SP2 instalado (servidor BT) - Para ficar disponível para ser encontrado por outros dispositivos BT foi necessário configurar as suas propriedades de rede BT;
- e um adaptador do tipo USB, o Conceptronic CBT200U2 V2.0 (ver Figura 6.3), funcionando com os *drivers* do Windows. Inicialmente, foi adquirido um outro adaptador (Anycom Bluetooth USB-Adapter USB-250) que obrigava à utilização dos próprios *drivers*.



Figura 6.3: directInfo2u - Dispositivo de Rastreo: O adaptador USB de Bluetooth utilizado.

### 6.4.1.2 Middleware

Este módulo da infra-estrutura assenta na camada intermédia da arquitectura presente na Figura 6.1. Esta camada foi implementada com recurso ao uso do **Bluetooth Stack da Microsoft** (MS BT Stack), que se encontra no sistema operativo instalado. O MS BT Stack é independente da tecnologia de programação, sendo o responsável pela recepção dos dados a partir da camada de *hardware* - Dispositivo de Rastreo - e pela entrega dos mesmos à camada superior. Os *drivers* do Windows referidos na subsecção anterior são, efectivamente, o MS BT Stack, que implementa os protocolos superiores do BT. É através deste Stack que se controla o dispositivo de BT. Duas das alternativas mais comuns são o BlueSoleil<sup>4</sup> e o WIDCOMM<sup>5</sup>.

As aplicações implementadas em plataforma Java acedem ao dispositivos BT através do pacote JABTW<sup>6</sup> (*Java APIs for Bluetooth Wireless Technology*). Contudo, este pacote é definido como **JSR-82**<sup>7</sup> [150] no interior do *Java Community Process*<sup>8</sup> (JCP). A especificação desta API foi finalizada em 2002, mas apenas recentemente é que tem vindo a ser ampla e livremente utilizada em aplicações de BT. A JSR-82 esconde a complexidade do *stack* de protocolos BT, disponibilizando um conjunto simples de API's Java para o desenvolvimento, que permitem, por exemplo, gerir os parâmetros do dispositivo BT local, encontrar outros dispositivos, procurar por serviços disponíveis nesses dispositivos, ligar a um serviço e comunicar, registar um serviço, e gerir mecanismos de segurança.

Contudo, a JSR-82 foi originalmente desenvolvida para dispositivos móveis, e mais concretamente para a plataforma J2ME. Assim sendo, teve que ser encontrada uma implementação da JSR-82 para J2SE. Analisando o mercado, podem ser encontrados dois tipos de implementações: um baseado num stack nativo que é envolvido por uma interface Java, e outro que é completa-

---

<sup>4</sup><http://www.bluesoleil.com/>

<sup>5</sup><http://www.widcomm.com/>

<sup>6</sup>O JABTW também é conhecido por BTAPI.

<sup>7</sup>A API JSR-82 encontra-se em: <http://jcp.org/aboutJava/communityprocess/final/jsr082/>

<sup>8</sup><http://jcp.org/en/home/index>

mente implementado em Java, sendo descrito como independente de plataforma. Na Figura 6.4 pode ser observada uma tabela que apresenta as características principais de um conjunto de implementações da API JSR-82.

Nome	Suporta javax.bluetooth	Suporta javax.obex	Java Plataformas	Sistemas Operativos	Licença	Preço	Comentários
<a href="#">Atinav</a>	Sim	Sim	J2ME, J2SE	Win-32, Linux, Pocket PC	Comercial	Não especificado	
<a href="#">Avetana</a>	Sim	Sim	J2SE	Win-32, Mac OS X, Linux, Pocket PC	Comercial	€25 Por licença ilimitada	Necessita Stack Bluetooth externa
<a href="#">BlueCove</a>	Sim	Não	J2SE	WinXP SP2	GNU	Grátis	Suporta L2CAP e RFCOMM. Necessita Stack Bluetooth externa
<a href="#">Harald</a>	Não	Não	Qualquer plataforma que suporte javax.comm	Maior Parte	Livre	Grátis	Suporta apenas L2CAP. Implementação interna da stack Bluetooth
<a href="#">JavaBluetooth</a>	Sim	Não	Qualquer plataforma que suporte javax.comm	Maior Parte	GNU	Grátis	Suporta L2CAP. Implementação interna da stack Bluetooth
<a href="#">JBlueZ</a>	Não	Não	J2SE	Linux	GNU	Grátis	Suporta apenas descoberta básica. Necessita Stack Bluetooth externa
<a href="#">Impronto DK</a>	Sim	Sim	J2SE	Linux	Comercial	Não especificado	Grátis para uso educacional. Necessita Stack Bluetooth externa
<a href="#">aveLink</a>	Sim	Sim	Todos	Todos	Comercial	Não especificado	Implementação interna da stack Bluetooth

Figura 6.4: directInfo2u - Middleware: Tabela comparativa com as implementações da API JSR-82 para J2SE.

Para este protótipo foi escolhida a **BlueCove** [12] por ser *open source*, apresentar muita documentação e exemplos de utilização, e assentar no MS BT Stack (também funciona, por exemplo, com o BlueSoleil ou com o WIDCOMM). A BlueCove foi inicialmente desenvolvida como projecto pela Intel Research e actualmente é mantida por voluntários. A API não implementa a JSR-82 por completo, mas implementa os perfis essenciais para os protocolos SDP, RFCOMM e L2CAP.

Por outro lado, a BlueCove utiliza a interface **JNI**<sup>9</sup> (*Java Native Interface*) como tradutor entre o MS BT Stack e a sua própria implementação do *stack*.

Assim sendo, este módulo Middleware é, essencialmente, composto pelo MS BT Stack e pela API BlueCove (ver Figura 6.2).

<sup>9</sup><http://java.sun.com/developer/onlineTraining/Programming/JDCBook/jni.html>

6.4.1.3 Gestor de Rastreo

Este módulo utiliza o Middleware, e mais concretamente a interface BlueCove, para fazer o rastreo de dispositivos móveis que se encontram na zona de implementação. Observando novamente a Figura 6.2, verifica-se que este módulo já se encontra na camada designada por "Aplicação em J2SE", sendo o responsável por fazer a interface entre o núcleo da aplicação específica e o conjunto *hardware/APIs* utilizado. Funciona como API direccionada para o tipo de infra-estrutura proposta, disponibilizando os dados de rastreo de forma normalizada para o componente applicacional. Assim sendo, os dados são fornecidos numa mensagem formatada em XML, tal como acontecia com o primeiro protótipo.

Uma aplicação que lida com Bluetooth segue os passos seguintes [13]:

1. Iniciação do *stack*: Este é, obrigatoriamente, o primeiro passo a ser dado. Consiste num certo número de pontos, mas, na essência, tem como principal propósito preparar o dispositivo (o servidor BT) para a comunicação BT. Este componente é dependente da API que implementa a JSR-82.
2. Descoberta de dispositivos: A detecção dos dispositivos que se encontrem na zona de implementação é feita utilizando a classe *DiscoveryAgent* e a interface *DiscoveryListener*.

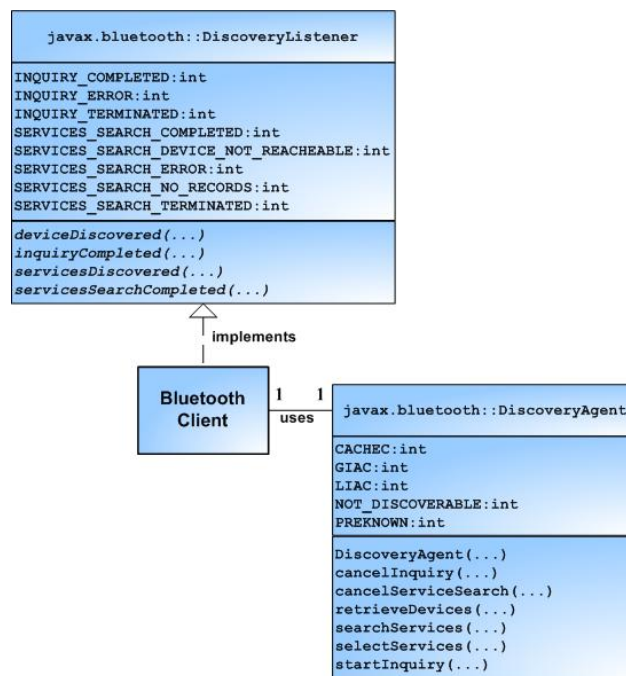


Figura 6.5: directInfo2u - Gestor de Rastreo: Pacote JSR-82 para procurar dispositivos [13].

3. Gestão de dispositivos: Esta gestão é feita através de duas classes principais presentes

na API: LocalDevice e RemoteDevice. Estas classes permitem retirar algumas informações tanto do dispositivo local, como dos dispositivos remotos encontrados na zona. Por exemplo, através da segunda consegue-se saber o endereço de cada um dos dispositivos remotos. Este endereço funciona como um endereço MAC da placa de rede de um computador. Cada dispositivo BT possui um endereço único de 48 bits (BD\_ADDR), obtido a partir da IEEE Registration Authority. O endereço é uma *string* com valores em hexadecimal e composta por 12 caracteres, aparecendo com o carácter ':' a separar os valores (e.g., 00:17:e3:8e:b9:2a).

4. Descoberta de serviços: Este passo é essencial quando não se sabe o que pode ser oferecido pelo dispositivo descoberto. Funciona tal como a descoberta de dispositivos, usando-se novamente a classe DiscoveryAgent na procura. A classe apresenta métodos para o efeito (ver Figura 6.5). Por exemplo, o método servicesDiscovered() será chamado pela JVM (Java Virtual Machine), passando um objecto do tipo ServiceRecordobject. Estes objectos são registados numa base de dados especial disponibilizada pela API, designada por *Service Discovery DB* (SDDB). Um serviço é identificado por um valor único (UUID) de 128 bits, mas que pode aparecer apenas com 32 ou 64.
5. Comunicação: Após a descoberta e iniciação dos serviços pretendidos pode ser feita a comunicação. Existem três opções para se fazer a comunicação: L2CAP, RFCOMM, e OBEX. Por exemplo, utilizando RFCOMM, o endereço URL da ligação cliente SPP (*Serial Port Profile*) teria a *string* "btspp://0017E38EB92A:3", sendo esta composta pelo BD\_ADDR do dispositivo remoto e pelo número de canal (3) atribuído ao serviço quando foi registado na SDDB. Na comunicação, o dispositivo 0017E38EB92A é o servidor e o que fez a ligação é o cliente (e.g., o servidor BT do sistema implementado). Neste tipo de comunicação, um dispositivo pode alternar facilmente entre ser cliente e ser servidor.

O presente módulo Gestor disponibiliza uma classe (Rastreio) que integra os primeiros três componentes. Os restantes são implementados à parte, no sub-módulo ICCA para a comunicação de curto alcance.

Para o presente cenário, esta API inicia o Stack, faz o rastreio e retira informação sobre o dispositivo detectado. Sempre que um dispositivo surge ("*observed*") ou desaparece ("*lost*") é criada uma mensagem XML, que contém a data e hora do evento, as identificações do dispositivo remoto (a *tag* é o BD\_ADDR nesta aplicação) e do dispositivo leitor (o servidor BT) (ver Figura 6.6).

Ao contrário do protótipo anterior, no qual o BM utilizava métodos da API que lhe davam

#### 6.4. INSTANCIACÃO DA ARQUITECTURA

---

a indicação de ocorrências dos eventos "observed" e "lost" (parametrizados no RFID Anywhere), para a presente aplicação com BT esses eventos tiveram que ser programados neste módulo. Deste modo, para um ciclo de procura, estabeleceu-se o seguinte procedimento:

- Criar uma mensagem XML com *eventType*="observed", por cada dispositivo que aparece na nova lista;
- Comparar a lista de dispositivos encontrados com a lista do ciclo anterior;
- Criar uma mensagem XML com *eventType*="lost", por cada dispositivo que aparece apenas na lista anterior.

Um dispositivo encontra-se presente na zona da aplicação enquanto não aparecer associado a um evento do tipo "lost". Assim sendo, se for observado em N ciclos consecutivos, então são N o número de mensagens XML criadas relativamente a ele. Caberá ao componente aplicacional tratar estes dados da forma que for mais conveniente para a aplicação.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <Detection eventType="observed">
3   <Source>servidorBT</Source>
4   <TagID>00:17:e3:8e:b9:2a</TagID>
5   <TagType>Computador Palm</TagType>
6   <Date>2007-06-10</Date>
7   <Time>15.46</Time>
8 </Detection>
```

Figura 6.6: directInfo2u - Gestor de Rastreo: Mensagem XML de uma detecção.

Enquanto que com a tecnologia RFID se obtinha directamente o momento em que uma *tag* aparecia ou desaparecia, com o BT é necessário programar-se um ciclo de procura para determinar quem está presente. Determinar um tempo de duração para este ciclo não é uma tarefa simples, pois por vezes há dispositivos que necessitam de um pouco mais de tempo. Assim sendo, programaram-se os ciclos com uma duração média de 2 segundos, obtendo-se bons resultados. Contudo, estes valores devem ser experimentados nas mais diversas situações e para o maior número de dispositivos a detectar.

As mensagens referentes a um ciclo de procura são disponibilizadas para o componente aplicacional através de uma lista.

É importante referir que a programação deste módulo requereu técnicas de programação concorrente. As operações relacionadas com procura e gestão dos dispositivos BT não poderiam ser feitas na *thread* da aplicação, uma vez que pararia em determinados momentos. A descoberta (e comunicação) de dispositivos teve que ser feita em *threads* separadas para também poderem

ser canceladas sempre que necessário.

### 6.4.2 Componente Aplicacional

Já foi referido que a linguagem de programação **Java** foi a utilizada para o desenvolvimento deste protótipo. A linguagem **C#** era também uma forte hipótese, mas o elevado grau de portabilidade, as APIs e bibliografia encontradas para a implementação de BT, e a oferta da plataforma J2ME foram determinantes para a escolha.

À semelhança do protótipo anterior, também este utiliza o **MySQL DB server**, como servidor gestor de bases de dados (SGBD), e a especificação **XML**, para o intercâmbio de documentos/mensagens entre módulos, pelas razões apontadas na subsecção 5.4.2.

De seguida são descritas as opções tomadas para cada um dos módulos e sub-módulos que compõem este componente.

#### 6.4.2.1 Módulos Personalização e Conteúdos de Objectos

Todas as opções tomadas para os módulos do protótipo *divingForPearls* correspondentes aos desta aplicação podem ser reutilizadas. Contudo, são de referir algumas especificidades resultantes da presente aplicação.

O módulo Personalização apresenta os perfis presentes na tabela 6.1. Estes perfis são suficientes para a realização de testes à infra-estrutura implementada, mas julga-se que, ao longo do tempo, venham a ser muitos os perfis criados. Certamente que surgirão alguns do cruzamento entre os existentes e outros aparecerão devido à diversidade de conteúdos requisitados pelos utilizadores das instalações. Tal como aconteceu com o primeiro protótipo, também aqui os dados dos utilizadores são privados, mas, com esta aplicação, os outros (que até podem ser estranhos à instituição) conseguem saber um pouco mais sobre o utilizador. Por exemplo, que arte pratica, com quem a pratica e em que condições.

Relativamente aos perfis, o perfil 0 é dedicado a todos os utilizadores que, por qualquer razão, preferem não ser introduzidos numa área específica. Por outro lado, os perfis do 1 ao 4 são para quem quer receber uma maior quantidade, em poucos segundos (enquanto passa pela sala de recepção), de informações sobre a sua área de interesse principal. Por exemplo, um aluno de TaiChi terá as últimas informações sobre as suas aulas, podendo estas servir de apoio a uma actividade do seu dia-a-dia. O último dos perfis (inexistente no cenário museu) é para todos os "visitantes" que, não estando registados, não são considerados utilizadores "oficiais" do sistema. Estes possuem dispositivos BT que são detectados, podendo visualizar algumas

<b>Perfil</b>	<b>Campos de Preferência</b>
0	Genérico - sem preferências
1	Aulas TaiChi
2	Aulas KungFu
3	Eventos e publicações
4	Instrutor
5	Não registado

Tabela 6.1: directInfo2u - Personalização: Perfis criados no módulo.

linhas de publicidade sobre a instituição. Este serviço será até mais interessante para a própria instituição, pois pode ser interessante guardar os dados relativos a estas detecções para serem feitos alguns estudos estatísticos. A frequência com que estes dispositivos interagem com o espaço pode ser um exemplo.

A gestão dos perfis foi feita através da aplicação para gestão de perfis criada para o protótipo anterior. Embora se esteja num cenário completamente diferente do anterior, o processo de gestão de perfis é realizado do mesmo modo.

Quanto ao módulo Conteúdos Objectos, este consiste numa base de dados que armazena, essencialmente, dados textuais. Mais uma vez, os dados relacionam-se com o objecto (único) e com o perfil. Contudo, também existem dados referentes aos utilizadores (que são membros da YMAA) e às actividades realizadas, pois este conhecimento será fundamental para disponibilizar a informação privada no dispositivo móvel.

### 6.4.2.2 Gestor e Fila de Eventos

Este módulo faz a gestão das mensagens XML recebidas do módulo de Rastreo, filtrando-as e organizando-as para o Interpretador Contextual (IC). A programação deste módulo deve ser parametrizada e alterada de acordo com condições a estabelecer para o funcionamento do sistema. Para a presente aplicação, fizeram-se algumas escolhas a título de teste.

A classe gestora possui uma lista auxiliar de dispositivos observados (`ObservedDevicesList`), onde são adicionados os novos dispositivos detectados e é contabilizado o número de ciclos consecutivos daqueles que se mantêm presentes. Um `ObservedDevice` é composto pelos atributos `BD_ADDR` e `obsNumber`. Ao receber uma lista de mensagens do módulo de Rastreo aplica o procedimento seguinte:

- Relativamente à lista ObservedDevicesList:
  - Colocar `obsNumber=11`, para todos os dispositivos associados a um `eventType="lost"`.
  - Adicionar à lista os dispositivos com `eventType="observed"`, que não faziam parte da mesma, e colocar `obsNumber=1`.
  - Actualizar `obsNumber=obsNumber+1`, para todos os dispositivos que faziam parte da lista.
  - Retirar da lista os dispositivos com `obsNumber=20`.
- Relativamente à lista de mensagens XML a passar ao IC:
  - Colocar as mensagens com `eventType = "observed"` para `"lost"`, caso `obsNumber >10`.
  - Ordenar a lista pelo atributo `obsNumber`, e por ordem crescente (o valor 1 significa que é o mais recente na zona).
  - Disponibilizar a lista como Fila de Eventos.

Em resumo, este procedimento faz com que um dispositivo que tenha sido detectado em dez ciclos consecutivos deixe de ser incluído nas observações passadas para o IC, ficando perdido ao atingir esse limite. É uma forma de gerir o tempo de apresentação no ecrã, sendo a parametrização do valor directamente dependente do tráfego de utilizadores em cada fase do dia. Um utilizador que esteja a ser detectado entre dez e vinte vezes consecutivas considera-se perdido. No entanto, ao atingir o número vinte passa ao estado em que pode voltar a ser observado.

### 6.4.2.3 Interpretador Contextual

Este módulo processa a lista de mensagens que se encontra na Fila de Eventos. Em termos lógicos, o processamento é dividido em duas partes.

Numa primeira, ao processar cada mensagem XML, com a indicação da detecção, faz a actualização imediata da sua base de dados (BD). A actualização passa por duas fases:

1. Pesquisar o dispositivo BT pelo seu `BD_ADDR`. Caso não exista na tabela Etiqueta (dispositivo "visitante" não registado) então é inserido na mesma. Também é inserido um Utilizador desconhecido e atribuído o perfil 5.
2. Efectuar o procedimento descrito para o IC do `divingForPearls` (ver Subsecção 5.4.2.3), para qualquer dispositivo.

Tal como acontecia com o IC do `divingForPearls`, a segunda parte do processamento de cada mensagem leva à criação de uma mensagem XML por objecto. Dado que neste cenário se

## 6.4. INSTANCIACÃO DA ARQUITECTURA

está perante um único objecto, é sempre criado apenas um documento XML com as todas as interacções que se encontram activas, sendo a saída para o módulo Gestor de Composição. A estrutura deste documento XML é exactamente a mesma que aparece na Figura 5.7.

### 6.4.2.4 Gestor de Composição

Relativamente ao módulo correspondente do protótipo anterior, o presente GdC também cria documentos XML com informação para ser consultada pelos dispositivos móveis, para além de efectuar tudo aquilo que foi descrito na subsecção 5.4.2.4.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <directinfo2u>
3   <data>31-08-2006</data>
4   <noticias_lista>
5     <noticia id="X"> <titulo>Novas Edicao</titulo><data_noticia>13-08-2006</data_noticia>
6     <conteudo>Lancamento do novo livro de Dr. Yang e Jwing-Ming intitulado "TAJIQUAN THEORY OF DR. YANG, JWING-MING THE
7     ROOT OF TAJIQUAN"</conteudo> </noticia>
8     <noticia id="Y"> <titulo>Novo artigo</titulo><data_noticia>15-08-2006</data_noticia>
9     <conteudo>Titulo: A Respiracao no Taijiquan, Autor: Dr. Yang Jwing Ming</conteudo></noticia>
10    </noticias_lista>
11    <aulas_lista>
12      <aula id="1" tipo_aula="Shaolin"><data_aula>18-09-2006</data_aula> <instructor>Dr. Yang</instructor><local_aula>Estadio Universitario
13      </local_aula>
14      <programa>
15        1. Si Lu Ben Zha (Quatro Modos de Correr e Esmagar) (Forma de Punho - Punho Longo)
16        2. Ba Mei Shou (Mao das Oito Plumas) (Forma de Punho - Grou Branco)
17        3. Qi Men Jian (Espada da Familia Qi) (Forma de Espada - Punho Longo)
18        4. Er Lu Qiang (Segunda Via da Lanca) (Forma de Lanca - Punho Longo)
19        5. Dan Dao Dui Qiang (Sabre contra Lanca) (Forma de Dois Homens Armas - Punho Longo)
20      </programa></aula>
21      <aula id="2" tipo_aula="Taijiquan"><data_aula>27-09-2006</data_aula><instructor>Tony Chee</instructor><local_aula>AIRFA Academia
22      Almadense - Sala 13</local_aula><programa> I. Forma de Punho, II. Treinos Especiais, III. Combate Livre de Taiji</programa></aula>
23    </aulas_lista>
24    <eventos_lista><proximo_evento><data_evento>27 de Marco a 2 de Abril</data_evento>
25    <descricao>Seminario Primavera</descricao><local_evento>Amadora</local_evento></proximo_evento>
26  </directinfo2u>
```

Figura 6.7: directInfo2u - Gestor de Composição: Um exemplo de documento XML para ser lido por um dispositivo móvel.

Sempre que recebe um documento a partir do IC, questiona o servidor de Conteúdos de Objectos para obter os dados considerados recentes, i.e., todos aqueles que se relacionam com o utilizador detectado, mas que foram inseridos após a última detecção do utilizador. Se existirem dados recentes então cria um novo documento XML, como o que pode ser visto na Figura 6.7, que disponibiliza ao módulo Gestor de Pedidos. Deste modo, um utilizador poderá ter acesso a informação privada, complementar da apresentada no ecrã público, ou até mesmo a que não poderia ser apresentada por existir um elevado número de utilizadores no espaço.

O documento é disponibilizado ao Gestor de Pedidos para que através deste possa chegar ao dispositivo móvel, mas como alternativa, caso este não implemente a API JSR-82, poderá

ser colocado no Servidor de Informação. À semelhança do que acontece com os ecrãs, neste poderá existir uma directoria para cada dispositivo móvel, que através de GPRS poderá pedir o documento mais recente.

#### 6.4.2.5 Gestor de Pedidos

Este módulo é o responsável pela gestão dos documentos XML que contêm informações que aumentam o objecto através do dispositivo móvel. Para esta gestão ser realizada foi utilizada a **eXist** [151], uma base de dados nativa de XML. Esta opção por este tipo de sistema deveu-se ao facto de ser necessário gerir os documentos de forma persistente. O facto da eXist ser Open Source, *freeware*, conceituada no meio da comunidade XML, e de fácil utilização contribuiu para a sua escolha em detrimento de outras opções.

Assim sendo, o GdC solicita que este módulo armazene os documentos criados, enquanto que os dispositivos móveis, via módulo ICCA do componente de Rastreio, solicitam que eles sejam retirados e disponibilizados.

#### 6.4.2.6 Servidor de Informação

Esta implementação é em tudo semelhante à encontrada na subsecção (ver 5.4.2.5) relativa ao servidor do primeiro protótipo. Para este servidor Web foi criada apenas uma directoria referente ao objecto YMAA, na qual é colocado o documento XML e onde já se encontra uma Applet (esta substitui o Shockwave Director na presente aplicação).

Este servidor permite ainda que os dispositivos móveis possam fazer pedidos dos ficheiros XML para visualizarem os conteúdos na aplicação móvel. Esta funcionalidade é importante para todos os que não implementam a API JSR-82 para comunicação directa com o servidor BT. Também é útil na medida em que permite que um utilizador possa aceder às informações mais recentes a partir de um local exterior ao ambiente de aplicação.

#### 6.4.3 Componente de Interface Ecrã

Este componente é idêntico ao seu correspondente presente no primeiro protótipo. Em termos das tecnologias utilizadas para a apresentação da informação, optou-se novamente pela base em XHTML, sendo a apresentação da informação integrada numa Applet Java. Pensou-se numa reutilização da implementação, com Shockwave Director, feita para o primeiro protótipo, mas preferiu-se testar o sistema com uma Applet. Uma vez que a informação para testes era

essencialmente do tipo textual, achou-se que o lançamento da Applet com os textos a deslocarem-se na vertical poderia ser mais rápido e até mais interessante para a aplicação em questão.

Uma **Applet** [152] é um programa Java que pode ser incluído numa página Web, tal como uma imagem. Assim sendo, quando a página que contém a Applet é carregada para o dispositivo cliente, o código da Applet também é transferido para a mesma máquina, sendo executado pelo JVM do *browser* cliente.

De seguida descrevem-se as opções tomadas para cada um dos módulos.

### 6.4.3.1 Dispositivo de Visualização e Gestor de Dispositivo

O dispositivo utilizado para os testes foi um monitor TFT/LCD, que ficou directamente ligado ao computador servidor BT do sistema.

As opções tomadas para o Gestor de Dispositivo são equivalentes às que foram apresentadas para o mesmo módulo do protótipo *divingForPearls* (ver Subsecção 5.4.3.2).

### 6.4.3.2 Apresentação e Composição de Apresentação

Nesta implementação, um CA (Composição de Apresentação) é composto por uma página XHTML e por uma Applet Java. Mais uma vez, a página Web serve essencialmente para integrar o filme para posterior visualização no *browser*. O único CA de ecrã (ficheiros XHTML e Applet) criado foi colocado na respectiva directoria no Servidor Web de Informação.

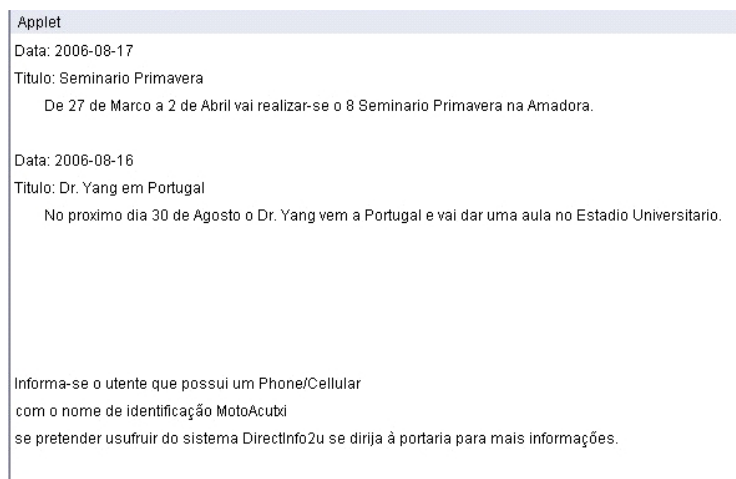


Figura 6.8: *directInfo2u* - Apresentação: Applet com informação genérica para utilizador registado e com indicação de como registar para um outro.

Tal como acontecia com o protótipo anterior, quando existem detecções, o Gestor de Com-

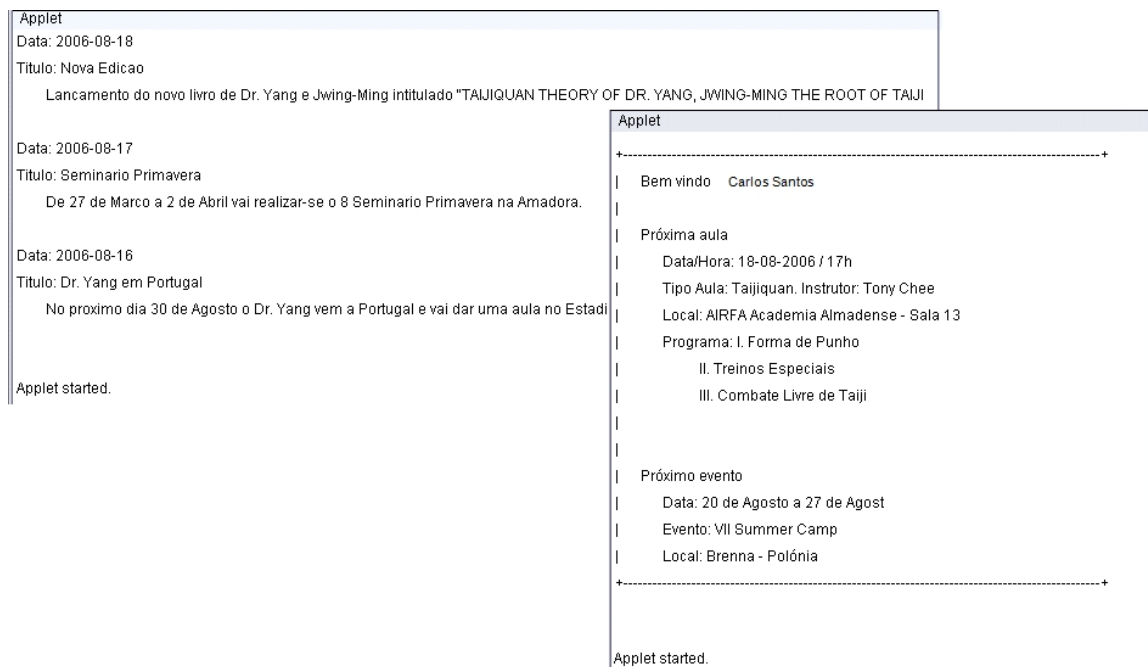


Figura 6.9: directInfo2u - Apresentação: (esquerda) Applet com informação genérica para dispositivo não registado e (direita) Applet para utilizador registado.

posição cria e coloca o documento XML na directoria do Servidor Web. Efectua o refrescamento automático da página Web, recarregando a Applet, que lê o novo ficheiro XML. Para esta implementação, o processamento do XML é feito através da biblioteca JAXP (*Java API for XML Processing*) [153].

#### 6.4.4 Componente de Rastreio - Módulo ICCA

O Interface de Comunicação de Curto Alcance (ICCA) é um módulo adicional (ver Figura 4.5) que apenas é implementado neste segundo protótipo, sendo responsável pela comunicação BT com os dispositivos móveis. Este módulo complementa o Gestor de Rastreio, implementando o último passo descrito em [13]. Pode ser ele a iniciar a comunicação com um dos dispositivos descobertos, mas para o presente protótipo ficou programado para aceitar os pedidos de comunicação RFCOMM vindos dos utilizadores.

Quando a aplicação cliente presente no dispositivo móvel solicita a comunicação, este questiona o módulo Gestor de Pedidos para obter o documento XML referente ao endereço do dispositivo. A comunicação não se estabelecerá caso não existam registos, pois significará que o dispositivo ainda não se encontra registado.

### 6.4.5 Módulo Aumentação Móvel

A Figura 6.10 mostra as camadas da arquitectura BT/Java para os dispositivos móveis, observando-se a separação lógica dos mesmos de acordo com os sub-módulos presentes no módulo Aumentação Móvel da arquitectura proposta.

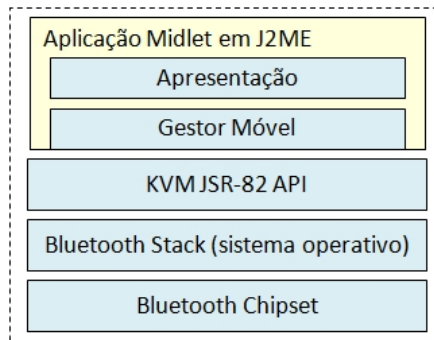


Figura 6.10: directInfo2u: Arquitectura Bluetooth/Java implementada para o dispositivo móvel.

Tal como já foi referido, a plataforma **J2ME** [14] foi adoptada para a programação da aplicação móvel. Esta plataforma direcciona a linguagem Java para os dispositivos móveis, possuindo as características de mobilidade, capacidade reduzida de processamento, pouca memória disponível e limitação de acesso à informação. Com o J2ME, é possível desenvolver, actualizar e instalar novas aplicações segundo as necessidades particulares de cada utilizador.

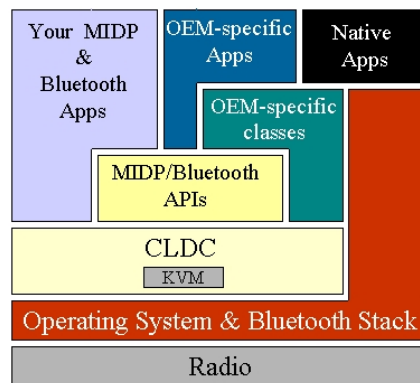


Figura 6.11: directInfo2u: Arquitectura CLDC/MIDP com integração de Bluetooth [14].

Para apoiar a larga gama de produtos que foram sendo criados dentro da extensão J2ME, foi introduzida a noção de configuração. Uma configuração define a plataforma Java para uma vasta gama de dispositivos, estando intimamente ligada à máquina virtual. Existem as configurações CDC (*Connected Device Configuration*) e CLDC (*Connected, Limited Device Configuration*), tendo sido esta última a seguida pelo protótipo, pois é direccionada para dispositivos com mais

limitações de memória e processador.

A camada superior da arquitectura J2ME é constituída por perfis. Pode-se considerar um perfil como uma extensão ao conceito de configuração, fornecendo as bibliotecas necessárias para a escrita de aplicações J2ME. O perfil seguido foi o MIDP 2.0 (*Mobile Information Device Profile*), que fornece as APIs necessárias para o desenvolvimento das aplicações móveis, chamadas de MIDlets. Na Figura 6.11 pode ser encontrada a arquitectura CLDC/MIDP com integração de Bluetooth.

De seguida são descritas as opções tomadas para cada um dos sub-módulos que compõem este componente.

#### 6.4.5.1 Tecnologia de Comunicação

A Tecnologia de Comunicação corresponde à camada *Bluetooth Chipset* da Figura 6.10. Hoje em dia, a maior parte dos dispositivos móveis pessoais já integram a tecnologia Bluetooth, mas os utentes do espaço que não possuam dispositivos destes não poderão ser utilizadores do sistema.

#### 6.4.5.2 Gestor de Comunicação

Esta camada funciona como módulo Middleware nos dispositivos móveis. O Gestor assenta nas camadas intermédias da arquitectura presente na Figura 6.10. O controlo do dispositivo é efectuado através do Stack BT que vem instalado no mesmo, e que a maior parte dos dispositivos já implementa. Por outro lado, ainda são poucos os que implementam a API JSR-82, necessária para se efectuar a comunicação BT através da aplicação MIDlet desenvolvida. Nos dispositivos móveis que a implementam, esta está directamente dependente do máquina virtual instalada. Uma lista de dispositivos móveis com suporte da API JSR-82 pode ser encontrada em [154].

#### 6.4.5.3 Gestor Móvel e Repositório RMS

O Gestor Móvel é a aplicação MIDlet que possibilita o aumento do objecto através da apresentação de informação adicional fora do cenário de aplicação. Este sub-módulo é o centro do módulo Aumentação Móvel. Esta aplicação permite que o utilizador solicite e receba a informação através de BT, caso o dispositivo implemente a API JSR-82. Em caso contrário, a aplicação continua a ser útil, pois existe a alternativa de questionar o Servidor de Informação e receber o documento XML via GPRS.

A comunicação BT entre o dispositivo móvel e o servidor BT pode ser efectuada por L2CAP, RFCOMM, ou OBEX. A última ainda não é implementada pela API BlueCove utilizada para

Nome	Tamanho	MIDP	Tipo
ASXMLP	6 kB	Sim	push, model
kXML 2.0	9 kB	Sim	pull
kXML 1.2	16 kB	Sim	pull
MinML 1.7	14 kB	Não	push
NanoXML 1.6.4	10 kB	Patch	model
TinyXML 0.7	12 kB	Não	model
Xparse-J 1.1	6 kB	Sim	model

Tabela 6.2: directInfo2u: *Parsers* XML para a plataforma MIDP.

o servidor BT, também não sendo necessária para o presente sistema, pois os dados são documentos XML enviados como *strings* de mensagens. Relativamente às duas primeiras, as ligações RFCOMM funcionam como canais de dados, enquanto que a L2CAP é orientada para os pacotes de dados. O protocolo RFCOMM é conhecido por implementar uma porta série virtual. Este foi o protocolo escolhido para a comunicação entre os dispositivos pois era o menos exigente a vários níveis, servindo perfeitamente para a transferência de mensagens.

Actualmente, estão a surgir *parsers* XML para a linguagem Java que já permitem aproveitar as capacidades do XML em ambiente MIDP. Estes *parsers* costumam necessitar de recursos de processamento e memória elevados relativamente ao ambiente em que se encontram. Para uma aplicação MIDP, os *parsers* XML devem ser pequenos, de forma a corresponder às restrições dos dispositivos móveis. A tabela 6.2 apresenta um conjunto de *parsers* XML de pequenas dimensões que são apropriados para MIDP.

Para a utilização neste protótipo foi seleccionado o kXML [155], pois verifica-se que é um dos *parsers* com menor tamanho, e sendo um *pull-parser* (*parsers* indicados para o processamento de ficheiros XML em dispositivos com recursos limitados) pode ser utilizado sem ser necessário qualquer tipo de alteração.

Por último, o Repositório RMS (*Record Management System*) é responsável pela persistência dos dados recebidos. Após efectuar o *parsing* da mensagem ou documento XML recebido, o Gestor armazena os dados no sistema RMS do MIDP. Deste modo, o utilizador terá acesso à informação de forma persistente e mesmo que se encontre fora do espaço de aplicação. O RMS, ou gestão de registos persistentes, suporta a criação de múltiplos *RecordStores*, que podem ser vistos como bases de dados, nas quais o *Record* é o elemento central.

## 6.4.5.4 Apresentação

Este módulo corresponde à interface visual com o utilizador, fornecendo um menu a partir do qual se pode escolher o tipo de informação a visualizar (ver Figura 6.12, imagem da esquerda). As opções são de acordo com os tipos das informações mais recentes disponíveis para o utilizador em questão.

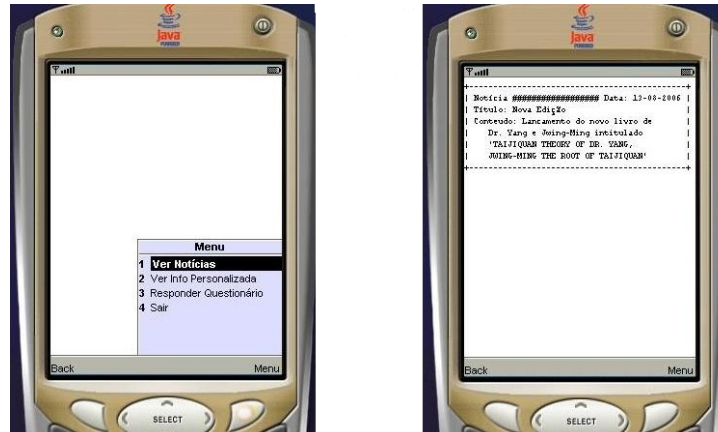


Figura 6.12: directInfo2u - Aplicação Móvel: (esquerda) O menu para acesso a informação privada e (direita) leitura de informação recebida (simulado no J2ME Wireless Toolkit).

Apesar de este módulo de Aumentação Móvel exigir a interacção activa por parte do utilizador, que inicialmente tem uma opção para requisitar informação ao sistema, tentou-se que fosse o menos complexo e intrusivo possível, limitando-se a apresentar a informação.

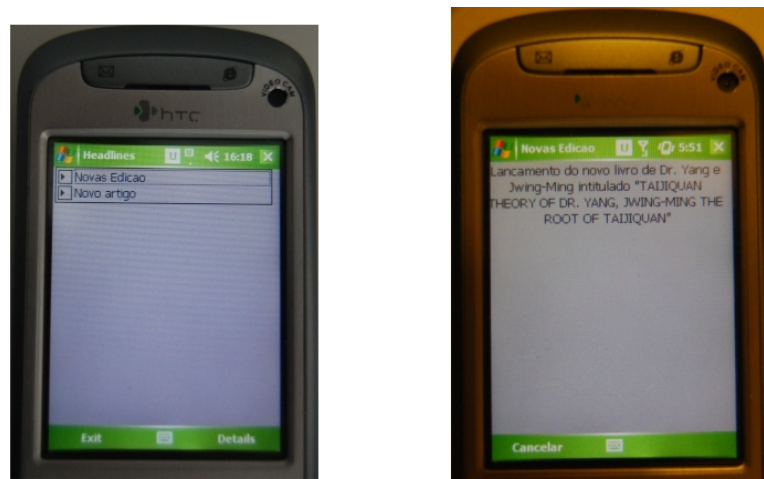


Figura 6.13: directInfo2u - Aplicação Móvel: Um outro exemplo em que o utilizador é actualizado com notícias sobre dois itens (visualizado num HTC TyTn).

### 6.4.6 Configuração Final

Sugere-se uma solução que inclua apenas um ecrã posicionado numa zona transiente da sala de aplicação. A configuração pode ser composta pelos componentes seguintes:

- 1 servidor BT, com adaptador BT tipo USB;
- 1 servidor Web (Servidor de Informação);
- 1 servidor de dados (módulos de Conteúdos e Personalização);
- Dispositivos móveis pessoais com tecnologia BT;
- 1 grande ecrã (de preferência, o maior possível);
- Rede LAN/WAN (TCP/IP).

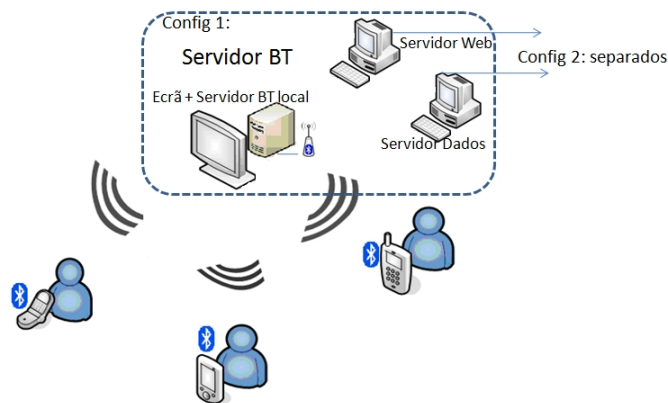


Figura 6.14: directInfo2u: Arquitectura de configuração física final.

Numa primeira fase, pode existir apenas um computador, designado por servidor BT, que hospeda os vários servidores, tendo uma ligação directa para o ecrã (monitor). Contudo, de futuro, poderão existir outros ecrãs espalhados pelas instalações, sendo assim aconselhável que os outros servidores passem para outras máquinas dedicadas aos respectivos serviços. A Figura 6.14 apresenta os três servidores integrados no computador servidor BT (Config 1), mas também indica a separação dos mesmos numa configuração alternativa (Config 2).

## Capítulo 7

# Conclusões e Trabalho Futuro

Neste capítulo, analisam-se os resultados obtidos, e as principais contribuições da tese, e indicam-se os pontos que devem ser abordados numa investigação futura.

Esta dissertação considerou uma perspectiva de aproximação prudente e iterativa à visão original de Weiser sobre a Computação Ubíqua. Recentemente, a investigação nesta área tem-se intensificado, mas o quotidiano humano ainda está longe de ser preenchido por sistemas inteligentes, invisíveis e não intrusivos. A ausência de uma infra-estrutura comum, que sirva de base de desenvolvimento, tem sido apontado como um obstáculo determinante à implementação de sistemas sólidos e viáveis nesta área. É natural que as especificidades de cada espaço e tipo de aplicação obriguem a diferentes desenhos de infra-estruturas. Assim sendo, com base na perspectiva pretendida, foi estudada e apresentada uma infra-estrutura direccionada para um tipo de aplicação bem definido, e com capacidade para ser gradualmente estendida.

Um dos objectivos do trabalho foi aproveitar a ubiquidade de ecrãs públicos e de dispositivos móveis, integrando-os nos princípios e no desenho da arquitectura. A definição da infra-estrutura centrou-se num tipo de aplicação, no qual os utilizadores podem ser discretamente detectados num espaço, onde um objecto é automaticamente aumentado através de informação apresentada num ecrã público. O objecto considerado podia ser o próprio espaço, desde que a sua informação fosse contextual, baseando-se na sua localização específica, no espaço e no perfil do utilizador detectado, sendo direccionada para este último.

Um outro objectivo importante foi mostrar que a implementação, de uma infra-estrutura computacional como a apresentada, podia ser conseguida à base de tecnologias maduras e correntes. Foram desenvolvidos dois protótipos que permitiram que as tecnologias fossem testadas, servindo como provas de viabilidade da infra-estrutura.

## 7.1 Resultados Obtidos

A resposta ao objectivo principal deste trabalho obrigou ao estudo da área científica em que este se enquadra, tendo permitido identificar as características que podem ser encontradas em aplicações de Computação Ubíqua. Assim sendo, verifica-se que características como a ubiquidade, a descentralização, a mobilidade, a conectividade *wireless*, a interconectividade, a portabilidade ou a centralização no utilizador são transversais a todos os sistemas de Computação Ubíqua, tendo sido fortemente consideradas no trabalho realizado.

Juntamente com estas características, ainda no capítulo dois, fez-se um levantamento dos elementos tidos como chave para a implementação de sistemas na área. Verifica-se que a localização é um elemento que tem vindo a ter uma importância crescente no dia-a-dia humano, como se constata através do mercado de dispositivos GPS. A identificação também é chave, relacionando-se de perto com o anterior. Este elemento é importante para que, num ambiente inteligente e volátil, as várias entidades possam ser reconhecidas através de uma identidade única. Já a conectividade é considerado um elemento central, agregador, devendo ter como prioridade as tecnologias *wireless*, pois estas são indicadas para os dispositivos portáteis/móveis. Tecnologias para conectividade também são usadas para identificação e localização. Um dos principais objectivos, senão o mais importante, nestes sistemas é o de facultar informação relevante, sempre que necessária, num determinado momento e num dado local. Um sistema inteligente deve adaptar e disponibilizar automaticamente a informação de acordo com o contexto de inserção. Conclui-se que a noção do contexto (Informação Contextual) é um outro elemento-chave. Por outro lado, cada vez mais, a monitorização sensorial possibilita que um ambiente de Ubicomp detecte e responda a situações, sem que para tal tenha que existir uma interacção humana propositada. Por último, constata-se que é fundamental que se adeque a interacção HCI aos novos sistemas discretos ubíquos, definindo-se para tal um novo paradigma.

Com vista a implementar protótipos da infra-estrutura proposta nesta dissertação, no capítulo três foram estudadas várias tecnologias que se relacionam com os elementos abordados.

Conclui-se que os contínuos avanços na Computação Embutida têm apresentado as tecnologias que mais possibilitam um avanço real na realização dos vários tipos de aplicações de Computação Ubíqua, tendo responsabilidade no sucesso das restantes. Conclui-se que a comunicação *wireless* mais interessante para a implementação dos sistemas pretendidos é a baseada em radiofrequência em detrimento dos infravermelhos, que necessitam de um alinhamento directo entre emissor e receptor para obterem resultados adequados. A radiofrequência apresenta me-

lhores taxas de transmissão e, no caso da preferida Bluetooth, também já pode ser encontrada na larga maioria dos dispositivos móveis pessoais do quotidiano. A tecnologia Bluetooth permite o estabelecimento de redes pessoais (PAN) oferecendo uma comunicação de curto alcance, que é um dos objectivos pretendidos. Esta tecnologia serve ainda para a identificação unívoca das entidades quando entram numa zona restrita de outra entidade. Referência ainda para a NFC, cujo alcance muito reduzido oferece um maior grau de segurança, sendo adequada para ambientes de multidões e sempre que o utilizador deva sentir consciência e controlo sobre os processos. Sendo compatível com as estruturas RFID, pode vir a ser importante para o estabelecimento de novos serviços agregados à infra-estrutura proposta. Verificou-se que a tecnologia RFID seria a ideal para localizar e identificar entidades num determinado ponto de um espaço. A precisão e detalhe com que se pode parametrizar e o facto de poder ser pouco intrusivo são mais valias para o sistema. Tecnologias como RTLS e GPS são direccionadas para se fazer o seguimento mais preciso do utilizador ao longo de vários espaços, interiores ou exteriores. Já os sensores são determinantes para aplicações, como por exemplo na área da saúde, nas quais é importante adquirir automaticamente dados relativos a temperatura, a pressão, entre outros. Hoje em dia, já existem várias plataformas de desenvolvimento, muito direccionadas para a área das redes de sensores, que facilitam o desenvolvimento de ambientes inteligentes. Por outro lado, a ubiquidade e a capacidade de integrar várias tecnologias dos dispositivos móveis são características essenciais para a infra-estrutura apresentada. Também foram estudadas várias tecnologias para ecrãs, podendo concluir-se que as soluções para disponibilização de informação embutida nos espaços públicos são muitas e diversas, tanto indicadas para aplicações discretas como pró-activas.

Após o estudo anterior, foi feito um levantamento de projectos e sistemas que apresentam pontos em comum com o presente trabalho (ver Secção 3.8). Contudo, este último diferencia-se deles em questões fundamentais, que se referem de seguida.

Inicialmente, é importante concluir quais, de entre os projectos apresentados, se baseiam nos princípios defendidos no capítulo quatro. Assim sendo, o BluBoard, o Hermes, o MobiLenin, o iRemote e o eCampus distanciam-se bastante, baseando-se, na essência, em interacções entre ecrãs públicos e dispositivos móveis pessoais, estando a pró-actividade inicial do lado do utilizador. Também parecem apresentar aplicações muito específicas e não têm o intuito de aumentar directamente um objecto, exceptuando o eCampus, que apresenta uma infra-estrutura mais genérica e visa aumentar um objecto, dando outras soluções para extensão. Este último tem como objectivo central a disponibilização de informação adicional sobre a instituição de aplicação.

Em comparação com o presente trabalho, apenas não faz a detecção da presença de utilizadores junto aos ecrãs para apresentar informação personalizada. No entanto, também apresenta uma infra-estrutura robusta e adaptável, sendo interessante o uso das *visual tags* para disponibilizar serviços informativos adicionais. O BluBoard funciona como um serviço com o qual o utilizador pode consultar a "sua" informação quando se depara com um ecrã público, mas os restantes já apresentam aplicações com uma vertente social.

Um outro projecto com uma vertente social é o GroupCast, mas que já se aproxima mais dos princípios pretendidos. Este sistema "sente" discretamente a co-presença dos utilizadores junto ao ecrã público, e mostra a informação de acordo com o perfil destes. Dois projectos que apresentam infra-estruturas mais completas são o Cooltown/Rememberer e o ActivitySpot. Ambos aumentam um objecto com informação e serviços adicionais, mas não fazem a detecção automática e discreta dos utilizadores. O primeiro é interessante para esta tese por apresentar uma implementação num museu, mas não utiliza ecrãs. Já o segundo, utiliza ecrãs e adequa-se ao perfil dos utilizadores. Ambos permitem uma interacção activa dos utilizadores através de dispositivos móveis.

O Wi-Art também tem uma implementação em museu, mas não disponibiliza a informação num ecrã público partilhado, disponibilizando-a no dispositivo móvel do utilizador, que detecta o objecto através de RFID. Aqui o paradigma de detecção é algo diferente do pretendido, pois é o utilizador que detém os meios para detectar os objectos. Já o magicmirror apenas não disponibiliza um nível de interacção activa através de dispositivos móveis, pois tem uma base muito similar à do tipo de sistema proposto. De todos os modos, permite a interacção activa através de um ecrã *touch screen* e também, tal como o protótipo divingForPearls, utiliza a tecnologia RFID. Destes últimos projectos, apenas o segundo tem o objectivo de apresentar uma informação personalizada, no entanto, ambos aplicam praticamente todos os princípios requeridos por esta tese.

Finalmente, são apenas quatro os projectos que mais se aproximam do tipo de sistema proposto, pois apresentam características de sistema de informação ambiental, integram ecrãs públicos e dispositivos móveis no seu funcionamento, visam aumentar o objecto de aplicação com informação contextual, e usam as interacções acidentais de utilizadores para sentir a co-presença destes. Estes projectos são os seguintes: BluScreen, DIABlu, AWARE, e Augmented Knight's Castle (AKC). Também utilizam como tecnologia central uma das propostas desta tese, Bluetooth, no caso dos três primeiros, e RFID, no último. Contudo, todos eles apresentam diferenças importantes relativamente ao trabalho proposto. O AKC não aumenta o objecto

através de informação num ecrã, mas através de áudio. Também não pretende que a informação esteja de acordo com o perfil do utilizador, pois o que é realmente detectado são peças (objectos) junto a outros objectos. Este sistema é particularmente dirigido para aplicação num jogo, e não para suporte de uma actividade quotidiana. O AWARE faz a localização geral dos utilizadores no interior do cenário, e não junto ao(s) ecrã(s) (não é regra, mas também pode acontecer). Assim sendo, um ecrã não apresenta automaticamente informação dirigida para um utilizador. Este terá que utilizar uma aplicação para aceder à informação, podendo fazê-lo através de ecrã(s) ou dispositivos móveis. Quanto ao BluScreen, este não permite que um utilizador tenha um papel activo. Faz a detecção automática dos utilizadores através do BT dos dispositivos móveis, mas não apresenta informação de acordo com um perfil anteriormente registado. O perfil vai sendo definido pelo número de vezes, e respectivas durações, em que o utilizador é detectado no espaço de aplicação, pois a informação é do tipo publicitária. Por último, o DIABlu é talvez aquele que mais se aproxima do pretendido. As diferenças encontram-se no facto de, à partida, apresentar uma arquitectura menos versátil, pois é essencialmente dirigido para aplicações relacionadas com arte. Daquilo que se conhece, aumenta um objecto, mas enquanto obra de arte, não disponibilizando informação adicional que seja uma mais valia para o dia-a-dia do utilizador. Não tem perfis para os utilizadores, apresentando perfis construídos de acordo com, por exemplo, o número de dispositivos com BT detectados na área.

Projecto	SIA	Co-presença	Ecrã público	Disp. móvel	Info. sobre	Info. persona.	User activo	Tec. central	Apresenta Arq.
magicmirror (Thebigspace)	X	X	X		Obj.	Pode ser	Touch-screen	RFID	
ActivitySpot (U.Minho)		x	X	X	Obj.	X	Disp. móvel	várias	X
Remember (HP Labs)		x		X	Obj.		x	IR câmara	X
Wi-Art (U. La Sapienza)		X		X	Obj.			RFID	
AWARE (Bardram, H.)	X	X	X	X	Obj.		Disp. móvel	BT	X
DIABlu (E.Artes/UCP)	X	X	X	X	arte		Disp. móvel	BT	X
A. Knight's Castle (Playmobil)	X	X		X	Obj.		Disp. móvel	RFID	x
BluScreen (Sharifi)	x	X	X	X	Obj.			BT	
Infraestrutura apresentada	x	X	X	X	Obj.	X	Disp. móvel	RFID, BT, ...	X

Figura 7.1: Quadro comparativo de sistemas e projectos relacionados.

A Figura 7.1 apresenta uma tabela onde se comparam os princípios e os principais factores de implementação relativamente à infra-estrutura proposta e aos trabalhos que mais se aproximam da mesma. Conclui-se que o tipo de infra-estrutura proposta apresenta importantes diferenças relativamente aos projectos e sistemas conhecidos, indicando um novo caminho a ser seguido na área da Computação Ubíqua.

Das comparações efectuadas, retira-se que, de uma ou de outra forma, os princípios propostos têm vindo a surgir em trabalhos similares, mas não têm sido apresentados como um conjunto claro de linhas orientadoras para o desenvolvimento de um tipo de sistema específico. Contudo, os trabalhos relacionados também ajudam a perceber e a concluir que estes princípios são indicados para o tipo de infra-estrutura desejada. O critério elementar para aumentar os objectos passa pela apresentação de informação onde e quando é necessária, de forma o mais discreta possível, e com pró-actividade dos objectos. A este conjunto de princípios junta-se a ubiquidade de ecrãs e dispositivos móveis. Prova-se que os ecrãs são ideais para "comunicarem" às pessoas (os utilizadores) a inteligência dos objectos e espaços, pois são explícitos e comuns. Os dispositivos móveis são, hoje em dia, vulgares ferramentas do quotidiano que devem ser considerados. Numa abordagem complementar da infra-estrutura, esta permite uma interacção activa do lado do utilizador a partir, preferencialmente, dos dispositivos pessoais, servindo como extensão ao princípio elementar de interacção acidental. Conclui-se que esta solução deixa em aberto a possibilidade de adição de novos serviços numa aplicação, desde que não colida com as actividades diárias centrais do espaço.

Por outro lado, a arquitectura genérica desenhada permite a implementação da infra-estrutura em qualquer espaço físico. Estima-se que os componentes especificados, e que foram desenvolvidos nos protótipos, possam ser directamente reutilizados ou facilmente adaptados a projectos semelhantes, mostrando a viabilidade de extensão que a proposta apresenta na área científica. A arquitectura proposta é robusta e flexível para resistir a alterações que incluam novos conceitos ou melhorias. Apresenta ainda o ponto positivo de considerar a distribuição do sistema pelas várias entidades presentes no ambiente, sejam utilizadores, dispositivos ou objectos.

Para os protótipos desenvolvidos foram escolhidos dois cenários, mas são muitas as possibilidades para aplicação da infra-estrutura. Para uma das aplicações pretendia-se que o espaço apresentasse vários objectos e que não exigisse acções suplementares aos utilizadores. Para a outra, pretendia-se um teste num cenário mais aberto, permitindo adicionalmente que os utilizadores pudessem ter algum tipo de controlo. Após estas implementações, conclui-se que a infra-estrutura se mostrou adequada para qualquer um dos cenários. Os princípios complementaram-se

e prova-se que são vários os módulos da arquitectura que não necessitam de uma abordagem diferente, ou de ser reprogramados, ao passar-se de um cenário para o outro. Obviamente que a utilização de tecnologias diferentes entre protótipos obrigou a uma programação diferente, mas os procedimentos foram exactamente os mesmos na maioria dos módulos. As diferenças centraram-se em testes e funcionalidades diferentes. Aliás, conclui-se que para uma preparação adequada da infra-estrutura, o maior número de tecnologias, para rastreamento e comunicação, deve ser considerado para que módulos dependentes destas, como, por exemplo, o Gestor de Rastreo, integrem sub-módulos (APIs) para cada caso.

Quanto às tecnologias centrais utilizadas, a programação de um protótipo foi feita em C# e a do outro foi em Java. Conclui-se que qualquer uma das tecnologias pode ser utilizada para a programação destes sistemas, pois integram várias plataformas e bibliotecas que facilitam no desenvolvimento. No presente trabalho, foram utilizadas separadamente, mas módulos programados numa delas podem ser integrados num sistema baseado na outra. Apesar de qualquer uma delas poder ser utilizada com as tecnologias de identificação/comunicação escolhidas, verifica-se que existem pontos para os quais uma é mais indicada do que a outra, como é o caso da linguagem Java para os dispositivos móveis.

Relativamente às tecnologias de identificação/comunicação utilizadas, RFID e Bluetooth, conclui-se que a primeira pode ser mais indicada para os cenários mais discretos, menos intrusivos. Esta também é mais precisa na identificação, sendo indicada quando existem vários objectos/ecrãs no mesmo espaço, distanciados por poucos metros. Permite um maior detalhe na configuração da distância a que se pretende que um utilizador seja detectado, sendo também mais rápida. Contudo, é de aplicação mais restrita, pois é mais complexa, mais cara e menos popular. São poucos os dispositivos móveis (sobretudo comerciais) que possuem tecnologia RFID (na vertente NFC) ou que integram leitores, enquanto que a tecnologia Bluetooth está presente em praticamente todos os tipos de dispositivos, sobretudo os móveis pessoais. Assim sendo, conclui-se que a última é ideal para cenários como o apresentado no protótipo directInfo2u, pois nestes casos pretende-se, acima de tudo, aproveitar a ubiquidade dos dispositivos móveis pessoais e a presença de poucos ecrãs. Apesar de a tecnologia RFID não obrigar o utilizador a parametrizar/configurar o dispositivo para ser detectado, pois é uma simples *tag*, exige que tenha que o "adquirir" (através de um registo) propositadamente para a aplicação. Por outro lado, com Bluetooth, pode aproveitar a rotina quotidiana de levar consigo o dispositivo pessoal, mas já é obrigado a tê-lo correctamente configurado. Esta tecnologia pode ser mais facilmente adaptada às situações, sendo mais interessante pelos serviços adicionais que pode suportar. Contudo, há

que ter a consciência que a utilização das capacidades adicionais a torna mais intrusiva. Assim sendo, conclui-se que, dependendo do cenário e tipo de aplicação, ambas apresentam prós e contras e que são duas soluções válidas para o tipo de sistema apresentado por esta dissertação. Para algumas aplicações pode ser interessante conjugar as duas, tal como fizeram Salminen et al. em [156]. Será um sistema com maiores custos e com maior grau de complexidade, mas o sistema implementado poderá utilizar Bluetooth apenas para as interações activas e focar a utilização de RFID nas interações acidentais. Fazendo um cruzamento entre os dois protótipos desenvolvidos, verifica-se que a arquitectura proposta pode acondicionar facilmente esta configuração.

## 7.2 Contribuições

Com a presente dissertação, contribui-se para a área científica da Computação Ubíqua através do levantamento e apresentação de tecnologias e trabalhos desenvolvidos. Contribui-se ainda, para este conhecimento, com o estudo de princípios e de uma arquitectura genérica para aplicação de um sistema UbiComp bem definido, que auxilie no quotidiano humano.

Este trabalho validou a utilização de uma infra-estrutura computacional para o aumento de objectos através da disponibilização de informação adicional num ecrã público partilhado e de serviços em dispositivos móveis. Assim sendo, contribui-se com uma infra-estrutura que pode acomodar diferentes tecnologias e dispositivos e diversos modos de interacção com os objectos.

Outra das contribuições importantes foi provar que, hoje em dia, as tecnologias e as ferramentas, tais como os dispositivos móveis pessoais, necessárias para se alcançar um sistema destes já são amplamente utilizadas em sistemas computacionais, sendo ainda utilizadas pelas pessoas em larga escala.

## 7.3 Trabalho Futuro

Durante a realização deste trabalho, levantaram-se algumas questões que suscitaram interesse, quer seja por completarem o que foi feito, quer seja por direccionarem a infra-estrutura para outros cenários. Assim sendo, apresentam-se algumas linhas para trabalho futuro:

- Conduzir mais testes ao funcionamento dos protótipos desenvolvidos de forma a perceber a robustez, flexibilidade e adaptabilidade da arquitectura. Os testes efectuados tiveram um número reduzido de utilizadores, sendo necessário fazerem-se testes com um grande

número de utilizadores a interagir simultaneamente com um par objecto/ecrã.

- Conduzir testes de usabilidade distribuindo questionários pelos utilizadores. Certamente que as suas experiências ajudarão a confirmar os resultados obtidos e a sugerir melhorias. As respostas aos questionários também serão essenciais para outros pontos do trabalho futuro, tal como a definição de interfaces.
- Estudar formas de garantir aos utilizadores que as tecnologias utilizadas vão de encontro às suas necessidades de segurança e privacidade. Actualmente, levantam-se várias questões quanto à segurança e, sobretudo, privacidade que uma tecnologia como a RFID pode quebrar. Muitos dos receios são aprofundados por Albrecht e McIntyre em [157]. Relativamente a Bluetooth, entre outras, pode-se destacar um tipo de ataque chamado de *Bluesnarfing*, que passa pelo acesso não autorizado a dados presentes no dispositivo. Um estudo recente [158] dá a conhecer todas estas vulnerabilidades, ajudando a perceber o que pode ser feito para dar garantias aos utilizadores.
- Testar a implementação da infra-estrutura com outras tecnologias para os módulos de rastreio e de aumentação móvel. A tecnologia GPS poderá ser utilizada em cenários abertos, enquanto que, por exemplo, RTLS e triangulação Wi-Fi podem ser alternativas para um rastreio interior mais efectivo. Também poderá ser interessante dotar o dispositivo móvel de capacidade para ser ele a ler, através de NFC, *tags* RFID.
- Conduzir estudos relativos à personalização dos sistemas implementados. Existem algumas questões em aberto quanto a este tema, incluindo como capturar o perfil de um utilizador de acordo com o contexto de aplicação. Desenvolver o sistema para uma instituição que conhece os seus utilizadores (caso do directInfo2u), assim como apresentar perfis pré-definidos para que, no momento de registo, o utilizador se "encaixe" num (caso do divingForPearls), podem simplificar o processo. De todos os modos, certamente que para outros tipos de cenários os métodos para personalização terão que ser outros.
- Desenvolver estudos relacionados com as interfaces de visualização. Nestes estudos deve entrar novamente a questão da personalização, além da tecnologia utilizada para o ecrã. Num primeiro passo, o trabalho futuro pode passar pela utilização de grandes ecrãs no projecto directInfo2u. Ao utilizar um ecrã deste tipo, quais são os critérios para a construção da interface? Dependendo do número de utilizadores que se encontram na proximidade, como deve ser apresentada a informação? A interface deverá ter uma maior ou menor componente multimédia? Este é um ponto determinante em termos de trabalho futuro.
- Adicionar módulos à arquitectura para disponibilização de serviços adicionais relacionados

### 7.3. *TRABALHO FUTURO*

---

com o conceito de Memórias Digitais [159], uma vez que as interações registadas, e seus atributos, podem ser utilizadas em serviços, tais como um álbum de anotações.

# Bibliografia

- [1] Mark Weiser. Ubiquitous computing. <http://sandbox.xerox.com/ubicomp/>, 1996.
- [2] Bill Schilit, Norman Adams, and Roy Want. Context-aware computing applications. In *Proceedings of the IEEE 1st Workshop on Mobile Computing Systems and Applications (WMCSA)*, pages 85–90, Santa Cruz, CA, US, Dezembro 1994.
- [3] International Telecommunication Union (ITU). ITU Internet Reports 2005: The internet of things (ITU 2005 7th edition). Technical Report 7, International Telecommunication Union, Geneva, 2005.
- [4] Louise Barkhuus. Context information vs. sensor information: A model for categorizing context in context-aware mobile computing. In *Symposium on Collaborative Technologies and Systems*, pages 127–133, San Diego, CA, 2003.
- [5] AT&T Laboratories Cambridge. The active badge system. <http://www.cl.cam.ac.uk/research/dtg/attarchive/ab.html>, 2002.
- [6] Bluetooth SIG Inc. Bluetooth. <http://www.bluetooth.com/bluetooth/>, 2007 (Último acesso: Setembro 2007).
- [7] Jun Rekimoto and Yuji Ayatsuka. Cybercode: designing augmented reality environments with visual tags. In *DARE '00: Proceedings of DARE 2000 on Designing augmented reality environments*, pages 1–10, 2000.
- [8] Michael Rohs. Real-world interaction with camera-phones. In *2nd International Symposium on Ubiquitous Computing Systems (UCS 2004)*, pages 39–48, 2004.
- [9] HIT Lab, HIT Lab NZ, and ARToolworks Inc. ARToolKit website. <http://www.hitl.washington.edu/artoolkit/>, 2007 (Último acesso: Agosto 2007).

- [10] Enrico Rukzio, Sergej Wetzstein, and Albrecht Schmidt. A framework for mobile interactions with the physical world. In *Wireless Personal Multimedia Communications (WPMC)*, 2005.
- [11] RSC Sybase. Estado da arte em RFID. Technical report, Sybase Portugal - RFID Solutions Center, 2007.
- [12] BlueCove Team. BlueCove. <http://www.bluecove.org/>, 2007 (Último acesso: Agosto 2007).
- [13] C. Enrique Ortiz. Using the Java APIs for Bluetooth wireless technology, part 1 (API overview). <http://developers.sun.com/mobility/apis/articles/bluetoothintro/index.html>, Dezembro 2004.
- [14] Sun Developer Network (SDN). The Java ME platform: the most ubiquitous application platform for mobile devices. <http://java.sun.com/javame/index.jsp>, 2007 (Último acesso: Junho 2007).
- [15] Mark Weiser. The computer for the 21st century. *Scientific American*, 265(3):94–104, Setembro 1991.
- [16] Nishkam Ravi and Liviu Iftode. A note on pervasive computing. In Anthony LaMarca, Marc Langheinrich, and Khai Truong, editors, *Common Models and Patterns for Pervasive Computing Workshop At the 5th International Conference on Pervasive Computing (Pervasive2007)*, volume 4480 of *LNCS*, Berlin Heidelberg, New York, Maio 2007. Springer.
- [17] Stefan Agamanolis. Designing displays for human connectedness. In Kenton OHara, Mark Perry, Elizabeth Churchill, and Daniel Russell, editors, *Public and Situated Displays: Social and Interactional aspects of shared display technologies*. Kluwer, 2003.
- [18] Tim Paek, Maneesh Agrawala, Sumit Basu, Steve Drucker, Trausti Kristjansson, Ron Logan, Kentaro Toyama, and Andy Wilson. Toward universal mobile interaction for shared displays. In *CSCW '04: Proceedings of the 2004 ACM conference on Computer supported cooperative work*, pages 266–269, New York, NY, USA, 2004. ACM Press.
- [19] Harry Brignull and Yvonne Rogers. Enticing people to interact with large public displays in public spaces. In *Proceedings of INTERACT'03*, pages 17–25, Zurique, Setembro 2003.

- [20] Shahram Izadi, Harry Brignull, Tom Rodden, Yvonne Rogers, and Mia Underwood. Dynamo: a public interactive surface supporting the cooperative sharing and exchange of media. In *Proceedings of the 16th Annual ACM Symposium on User interface Software and Technology (UIST'03)*, Vancouver, Canada, Novembro 2003.
- [21] MIT Project Oxygen. MIT Project Oxygen: Overview. <http://www.oxygen.lcs.mit.edu/Overview.html>, 2007 (Último acesso: Setembro 2007).
- [22] John Barton and Tim Kindberg. The cooltown user experience. Technical Report 2001-22, HP Laboratories, Palo Alto, Calif., 2001.
- [23] João Pedro Sousa and David Garlan. Aura: An architectural framework for user mobility in ubiquitous computing environments. In Bosch, Gentleman, Hofmeister, and Kuusela, editors, *Proceedings of 3rd IEEE/IFIP Conference on Software Architecture*, pages 29–43, Montreal, Agosto 2002. Kluwer Academic Publishers.
- [24] Albrecht Schmidt, Matthias Kranz, and Paul Holleis. Embedded information. In *Workshop Ubiquitous Display Environments on UbiComp 2004*, Setembro 2004.
- [25] Rui Neves Madeira and Nuno Correia. Interaction between shared displays and mobile devices in an augmented objects framework. In *UBICOMM 2007 - International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies proceedings*, pages 189–194, Papeete, Polinésia Francesa, Novembro 2007. IEEE Computer Society.
- [26] Philip K. Dick. *Ubik*. Vintage (edição de 1991), 1969.
- [27] Genevieve Bell. No more sms from jesus: Ubicomp, religion and techno-spiritual practices. In P. Dourish and A. Friday, editors, *UbiComp 2006: Ubiquitous computing, 8th international conference proceedings*, volume 4206 of *LNCS*, pages 141–158. Springer, Setembro 2006.
- [28] Yvonne Rogers. Moving on from weiser’s vision of calm computing: Engaging ubicomp experiences. In P. Dourish and A. Friday, editors, *UbiComp 2006: Ubiquitous computing, 8th international conference proceedings*, volume 4206 of *LNCS*, pages 404–421. Springer, Setembro 2006.
- [29] mundoPT.com. Metade da população mundial tem telemóvel. <http://www.mundopt.com/n-metade-da-populacao-mundial-tem-telemovel-11067.html>, Julho 2007.

- [30] David Ley. Ubiquitous computing. *Becta's Emerging Technologies for Learning*, 2:64–79, 2007.
- [31] Mark Weiser. Ubiquitous computing. *IEEE Computer*, 26(10):71–72, Outubro 1993.
- [32] Vipul Hingne, Anupam Joshi, John Michopoulos, and Elias N Houstis. On the grid and sensor networks. In *IEEE Fourth International Workshop on Grid Computing (GRID2003)*, pages 166–175, Agosto 2003.
- [33] Nigel Davies, Adrian Friday, and Oliver Storz. Exploring the grid's potential for ubiquitous computing. *IEEE Pervasive Computing*, 3(2):74–75, 2004.
- [34] Geoff Coulson, Paul Grace, Gordon Blair, David Duce, Chris Cooper, and Musbah Sagar. A middleware approach for pervasive grid environments. In *Workshop on Ubiquitous Computing and e-Research*, Maio 2005.
- [35] Terry Payne and Ora Lassila. Semantic web services. In *Guest Editors' Introduction*. IEEE Intelligent Systems, 2004.
- [36] Ryusuke Masuoka, Bijan Parsia, and Yannis Labrou. Task computing - the semantic web meets pervasive computing. In *Proceedings of the 2nd International Semantic Web Conference 2003 (ISWC 2003)*, Sundial Resort, Sanibel Island, Florida, USA, Outubro 2003.
- [37] Harry Chen, Filip Perich, Tim Finin, and Anupam Joshi. Soupa: Standard ontology for ubiquitous and pervasive applications. In *International Conference on Mobile and Ubiquitous Systems: Networking and Services*, Agosto 2004.
- [38] Sachin Singh, Sushil Puradkar, and Yugyung Lee. Ubiquitous computing: connecting pervasive computing through semantic web. *Information Systems and E-Business Management*, 4(4):421–439, Outubro 2006.
- [39] M. Weiser, R. Gold, and J. S. Brown. The origins of ubiquitous computing research at PARC in the late 1980s. *IBM Systems Journal*, 38(4):693–696, 1999.
- [40] Jonathan Fildes. Satellite navigation is going places. *BBC News*, page <http://news.bbc.co.uk/2/hi/science/nature/6294287.stm>, Janeiro 2007 (Último acesso: Setembro 2007).

- [41] Geekzone. Bluetags to install world's first bluetooth tracking system. <http://www.geekzone.co.nz/content.asp?contentid=1070>, 2003.
- [42] Brygg Ullmer and Hiroshi Ishii. Emerging frameworks for tangible user interfaces. *IBM Systems Journal*, 39(3-4):915–931, 2000.
- [43] George W. Fitzmaurice, Hiroshi Ishii, and William A. S. Buxton. Bricks: Laying the foundations for graspable user interfaces. In *CHI'95: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 442–449. ACM Press / Addison-Wesley Publishing Co., 1995.
- [44] Hiroshi Ishii and Brygg Ullmer. Tangible bits: towards seamless interfaces between people, bits and atoms. In *CHI'97: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 234–241. ACM Press, Março 1997.
- [45] Thomas T. Hewett, Ronald Baecker, Stuart Card, Tom Carey, Jean Gasen, Marilyn Mantel, Gary Perlman, Gary Strong, and William Verplank. Curricula for human-computer interaction. Technical report, ACM Special Interest Group on Computer-Human Interaction (SIGCHI) Curriculum Development Group, <http://www.sigchi.org/cdg/>, 1996.
- [46] Gregory D. Abowd and Elizabeth D. Mynatt. Charting past, present, and future research in ubiquitous computing. *ACM Transactions on Computer-Human Interaction, Special issue on HCI in the new Millenium*, 7(1):29–58, Março 2000.
- [47] Michael Beigl, Hans-W. Gellersen, and Albrecht Schmidt. Mediacups: Experience with design and use of computer-augmented everyday artefacts. *Computer Networks, Special Issue on Pervasive Computing*, 35(4):401–409, Março 2001.
- [48] Jun Rekimoto. Pick-and-drop: A direct manipulation technique for multiple computer environments. In *ACM Symposium on User Interface Software and Technology*, pages 31–39. ACM Press, 1997.
- [49] Steinar Kristoffersen and Fredrik Ljungberg. Designing interaction styles for a mobile use context. In *Proceedings of 1st International Symposium Handheld and Ubiquitous Computing (HUC'99)*, pages 281–288, Setembro 1999.
- [50] Jun Rekimoto, Yuji Ayatsuka, Michimune Kohno, and Haruo Oba. Proximal interactions: A direct manipulation technique for wireless networking. In *Proceedings of INTERACT'03:*

*IFIP TC13 International Conference on Human-Computer Interaction*, Zurique, Suíça, Setembro 2003.

- [51] Guanling Chen and David Kotz. A survey of context-aware mobile computing research. Technical report, TR2000-381, Dartmouth College, Hanover, NH, USA, 2000.
- [52] Bill Schilit and Marvin Theimer. Disseminating active map information to mobile hosts. *IEEE Network*, 8(5):22–32, 1994.
- [53] Carlos Santos. Suporte a informação contextual em ambientes aumentados. Master’s thesis, Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, 2003.
- [54] Albrecht Schmidt, Kofi Asante Aidoo, Antti Takaluoma, Urpo Tuomela, Kristof Van Laerhoven, and Walter Van de Velde. Advanced interaction in context. In *Proceedings of First International Symposium on Handheld and Ubiquitous Computing (HUC’99)*, pages 89–110, Karlsruhe, 1999.
- [55] Chia-Hsun Jackie Lee, Jon Wetzell, Chiun-Yi Ian Jang, Yang-Ting Bowbow Shen, Ting-Han Daniel Chen, and Ted Selker. Attention meter: A vision-based input toolkit for interaction designers. In *Extended Abstracts of CHI 2006*, Montréal, Québec, Canada, Abril 2006.
- [56] Tim Kindberg and Armando Fox. System software for ubiquitous computing. *IEEE Pervasive computing*, 1(2):1536–1268, Janeiro-Março 2002.
- [57] N. Davies and H.-W. Gellersen. Beyond prototypes: Challenges in deploying ubiquitous systems. *IEEE Pervasive Computing*, 1(1):26–35, Janeiro-Março 2002.
- [58] Oliver Storz, Adrian Friday, and Nigel Davies. Towards ‘ubiquitous’ ubiquitous computing: an alliance with ‘the grid’. In *System Support for Ubiquitous Computing Workshop at UbiCOMP’03*, Seattle, USA, Outubro 2003.
- [59] Roy Want, Andy Hopper, Veronica Falcão, and Jonathan Gibbons. The active badge location system. *ACM Trans. Inf. Syst.*, 10(1):91–102, Janeiro 1992.
- [60] P. Hewkin. Smart tags-the distributed-memory revolution. *IEE Review*, 35(6):203–206, Junho 1989.

- [61] Mark Weiser. Some computer science issues in ubiquitous computing. *Special Issue, Computer-Augmented Environments, Communications of the ACM*, 36(7):75–84, Julho 1993.
- [62] Manuel Román, Christopher K. Hess, Renato Cerqueira, Anand Ranganathan, Roy H. Campbell, and Klara Nahrstedt. Gaia: A middleware infrastructure to enable active spaces. *IEEE Pervasive Computing*, 1:74–83, Outubro-Dezembro 2002.
- [63] Jason A. Brotherton. *Enriching Everyday Experiences through the Automated Capture and Access of Live Experiences - eClass: Building, Observing and Understanding the Impact of Capture and Access in an Educational Domain*. PhD thesis, Georgia Institute of Technology, Julho 2001.
- [64] Robert J. Orr and Gregory D. Abowd. The smart floor: A mechanism for natural user identification and tracking. In *Proceedings of the 2000 Conference on Human Factors in Computing Systems*, The Hague, Netherlands, Abril 2000.
- [65] Cory D. Kidd, Robert Orr, Gregory D. Abowd, Christopher G. Atkeson, Irfan A. Essa, Blair MacIntyre, Elizabeth D. Mynatt, Thad Starner, and Wendy Newstetter. The aware home: A living laboratory for ubiquitous computing research. In *Proceedings of the Second International Workshop on Cooperative Buildings*, Outubro 1999.
- [66] K. van Mensvoort. Datafountain- money translated to water. <http://www.koert.com/work/datafountain/>, 2005.
- [67] Mark Weiser and John Seely Brown. Designing calm technology. *PowerGrid Journal*, 1(1), Julho 1996.
- [68] Bonnie C. Baker. Wireless communication using the IrDA standard protocol. In *Analog Design Note ADN006*. Microchip Technology Inc, Janeiro 2004.
- [69] NTT Corporation and Waseda University. Market requirements for IrBurst. <http://www.irda.org/associations/2494/files/Publications/>, Setembro 2003.
- [70] A.C. Boucouvalas and V. Vitsas. 100 Mbit/s IrDA optical wireless throughput performance analysis. In *IEEE Proceedings Optoelectronics*, 2003.
- [71] ZENSYS Inc. Z-Wave, The wireless control language. <http://www.zen-sys.com>, 2007 (Último acesso: Setembro 2007).

- [72] Rafael Kolic. Ultra Wideband - the next generation wireless connection. *Technology@Intel Magazine*, Março 2004.
- [73] Intel Corporation. Intel and 802.11. [http://www.intel.com/standards/case/case\\_802\\_11.htm](http://www.intel.com/standards/case/case_802_11.htm), 2007 (Último acesso: Setembro 2007).
- [74] Walter S. Mossberg. New T-mobile phones use Wi-Fi networks that save on minutes. *AllThingsD.com*, Setembro 2007.
- [75] Bluetooth SIG. Bluetooth specification. Technical report, <https://www.bluetooth.org/>, 2007 (Último acesso: Setembro 2007).
- [76] ZigBee Alliance. Zigbee specification. Technical report, <http://www.zigbee.org>, 2006.
- [77] David Rua, Nuno Martins, Pedro Reis, and João Paulo Sousa. Interface USB para recolha de dados de sensores remotos utilizando ZigBee e IEEE 802.15.4. *Departamento de Engenharia Electrotécnica e de Computadores, Faculdade de Engenharia da Universidade do Porto, Portugal*, 2006.
- [78] NFC Forum. Near Field Communication. <http://www.nfc-forum.org/aboutnfc/>, 2007 (Último acesso: Setembro 2007).
- [79] Mary Catherine O'Connor. RFID middleware market set for growth, change. *RFID Journal*, page <http://www.rfidjournal.com/article/articleview/2719/1/1/>, Outubro 2006.
- [80] Peter Dana. Global Positioning System overview. [http://www.colorado.edu/geography/gcraft/notes/gps/gps\\_f.html](http://www.colorado.edu/geography/gcraft/notes/gps/gps_f.html), 2000 (Último acesso: Setembro 2007).
- [81] Gaetano Borriello, Matthew Chalmers, Anthony LaMarca, and Paddy Nixon. Delivering real-world ubiquitous location systems. *Communications of ACM*, 48(3):36–41, Março 2005.
- [82] Jeffrey Hightower and Gaetano Borriello. Location systems for ubiquitous computing. *IEEE Computer*, 34(8):57–66, Agosto 2001.
- [83] Netlink Group. Real time locating systems. *Technology White Paper*, Novembro 2006.
- [84] Michael Beigl, Albert Krohn, Tobias Zimmer, and Christian Decker. Typical sensors needed in ubiquitous and pervasive computing. In *Proceedings of INSS 2004*, 2004.

- [85] Volkan Isler, Sampath Kannan, and Kostas Daniilidis. Sampling based sensor-network deployment. In *Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Setembro-Outubro 2004.
- [86] Tiago Camilo, André Rodrigues, Jorge Sá Silva, and Fernando Boavida. Redes de sensores sem fios, considerações sobre a sua instalação em ambiente real. In *CSMU2006 - Conferência sobre Sistemas Móveis e Ubíquos*, Guimarães, Portugal, Junho 2006.
- [87] Ian Akyildiz, WellJan Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, 40(8):102–114, Agosto 2002.
- [88] Antonio G. Ruzzelli, Richard Tynan, Michael J. O’Grady, and Gregory M.P. O’Hare. Advances in wireless sensor networks. *Encyclopedia of Mobile Computing & Commerce*, 1, 2006.
- [89] Martin Leopold, Mads Bondo Dydensborg, and Philippe Bonnet. Bluetooth and sensor networks: A reality check. In *Proceedings of the First International Conference on Embedded Networked Sensor Systems*, 2003.
- [90] Matthias Ringwald, Mustafa Yucel, and Kay Romer. Interactive in-field inspection of WSNs (demo abstract). In *3rd European Workshop on Wireless Sensor Networks (EWSN 2006)*, Fevereiro 2006.
- [91] Saul Greenberg and Chester Fitchett. Phidgets: Easy development of physical interfaces through physical widgets. In *Proceedings of the ACM UIST 2001 Symposium on User Interface Software and Technology*. ACM Press, Novembro 2001.
- [92] Matthias Kranz and Albrecht Schmidt. Prototyping smart objects for ubiquitous computing. In *Workshop on Smart Object Systems In Conjunction with the Seventh International Conference on Ubiquitous Computing (UbiComp 2005)*, 2005.
- [93] Jan Beutel, Oliver Kasten, Friedemann Mattern, Kay Roemer, Frank Siegemund, and Lothar Thiele. Prototyping wireless sensor network applications with BTnodes. In *Proc. 1st European Workshop on Sensor Networks (EWSN 2004)*, volume 2920 of *Lecture Notes in Computer Science*, pages 323–338, Janeiro 2004.
- [94] Crossbow. Imote2 - high-performance wireless sensor network node. Technical report, [www.xbow.com/Products/Product\\_pdf\\_files/Wireless\\_pdf/Imote2\\_Datasheet.pdf](http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/Imote2_Datasheet.pdf), 2007 (Último acesso: Setembro 2007).

- [95] Esa Tuulari and Arto Ylisaukko-oja. Soapbox: A platform for ubiquitous computing research and applications. In *Pervasive '02: Proceedings of the First International Conference on Pervasive Computing*, pages 125–138. Springer-Verlag, 2002.
- [96] Hans Gellersen, Gerd Kortuem, Albrecht Schmidt, and Michael Beigl. Physical prototyping with Smart-Its. *IEEE Pervasive Computing*, 3(3):74–82, 2004.
- [97] Christian Decker, Albert Krohn, Michael Beigl, and Tobias Zimmer. The particle computer system. In *Proceedings of the ACM/IEEE Fourth International Conference on Information Processing in Sensor Networks*, 2005.
- [98] Sokwoo Rhee, Deva Seetharam, Sheng Liu, Ningya Wang, and Jason Xiao. iBeans: An ultralow power wireless sensor network. In *UbiComp 2003: Ubiquitous Computing, 5th International Conference*, pages 211–212, 2003.
- [99] Brendan O’Flynn, S. Bellis, K. Delaney, J. Barton, S. C. O’Mathuna, Andre Melon Barroso, J. Benson, U. Roedig, and C. Sreenan. The development of a novel miniaturized modular platform for wireless sensor networks. In *IPSN '05: Proceedings of the 4th international symposium on Information processing in sensor networks*, pages 49–55, 2005.
- [100] Holger Junker, Paul Lukowicz, and Gerhard Troster. PadNET: Wearable physical activity detection network. In *Proceedings of the Seventh IEEE International Symposium on Wearable Computers (ISWC'03)*. IEEE, 2003.
- [101] Ember Corporation. Ember. <http://www.ember.com/>, 2007 (Último acesso: Outubro 2007).
- [102] Tim Kindberg, John Barton, Jeff Morgan, Gene Becker, Debbie Caswell, Philippe Debaty, Gita Gopal, Marcos Frid, Venky Krishnan, Howard Morris, John Schettino, Bill Serra, and Mirjana Spasojevic. People, places, things: Web presence for the real world. In *Proceedings of IEEE Workshop on Mobile Computing Systems and Applications*, Monterey, California, USA, Dezembro 2000.
- [103] William G. Griswold, Patricia Shanahan, Steven W. Brown, Robert Boyer, Matt Ratto, R. Benjamin Shapiro, and Tan Minh Truong. ActiveCampus: Experiments in community-oriented ubiquitous computing. *Computer*, 37(10):73–81, 2004.
- [104] Geethapriya Venkataramani and Srividya Gopalan. Mobile phone based RFID architecture for secure electronic payments using RFID credit cards. In *Proceedings of the The Second*

*International Conference on Availability, Reliability and Security (ARES'07)*, pages 610–620, Washington, DC, USA, 2007. IEEE Computer Society.

- [105] Michael Rohs and Christof Roduner. Camera phones with pen input as annotation devices. In *Pervasive 2005 Workshop on Pervasive Mobile Interaction Devices (PERMID)*, pages 23–26, 2005.
- [106] R. Ballagas, M. Rohs, J. Sheridan, and J. Borchers. BYOD: Bring your own device. In *Proceedings of the Workshop on Ubiquitous Display Environments (UbiComp 2004)*, 2004.
- [107] Brad A. Myers, Jeffrey Nichols, Jacob O. Wobbrock, and Robert C. Miller. Taking handheld devices to the next level. *Computer*, 37(12):36–43, 2004.
- [108] Jeffrey Nichols, Brad A. Myers, Michael Higgins, Joseph Hughes, Thomas K. Harris, Roni Rosenfeld, and Mathilde Pignol. Generating remote control interfaces for complex appliances. In *UIST'02: Proc. of the 15th Annual ACM Symposium on User Interface Software and Technology*, pages 161–170. ACM Press, 2002.
- [109] Gottfried Zimmermann, Gregg Vanderheiden, and Al Gilman. Prototype implementations for a universal remote console specification. In *CHI'02: CHI'02 extended abstracts on Human factors in computing systems*, pages 510–511. ACM Press, 2002.
- [110] Greg Welch, Henry Fuchs, Ramesh Raskar, Herman Towles, and Michael Brown. Projected imagery in your "Office of the Future". *IEEE Computer Graphics and Applications*, 20(4):62–67, Julho-Agosto 2000.
- [111] David Molyneaux and Gerd Kortuem. Ubiquitous displays in dynamic environments: Issues and opportunities. In *Workshop on Ubiquitous Display Environments In conjunction with UbiComp 2004*, Setembro 2004.
- [112] MIT. DataWall: Seamless, full motion, ultra high resolution projection display. <http://vlw.www.media.mit.edu/groups/vlw/DataWall-overview.htm>, 2007 (Último acesso: Outubro 2007).
- [113] Kai Li, Han Chen, Yuqun Chen, Douglas W. Clark, Perry Cook, Stefanos Damianakis, Georg Essl, Adam Finkelstein, Thomas Funkhouser, Timothy Housel, Allison Klein, Zhiyan Liu, Emil Praun, Rudrajit Samanta, Ben Shedd, Jaswinder Pal Singh, George Tzanetakis, and Jiannan Zheng. Building and using a scalable display wall system. *IEEE Computer Graphics and Applications*, 20(4):29–37, Julho-Agosto 2000.

- [114] NCSA. NCSA visualization. <http://vis.ncsa.uiuc.edu/?content=about&subcontent=history>, 2007 (Último acesso: Setembro 2007).
- [115] vizgroup. High resolution tiled display wall - room 139 computer building. <http://gears.aset.psu.edu/viz/facilities/displaywall/>, 2007 (Último acesso: Agosto 2007).
- [116] Tat-Jen Cham, Rahul Sukthankar, James M. Rehg, and Gita Sukthankar. Shadow elimination and occluder light suppression for multi-projector displays. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '03)*, volume 2, 2003.
- [117] Jay Summet, Mathew Flagg, James M. Rehg, Gregory M. Corso, and Gregory D. Abowd. Increasing the usability of virtual rear projection displays. In *International Workshop on Projector-Camera Systems (PROCAMS)*, 2003.
- [118] Claudio S. Pinhanez. The everywhere displays projector: A device to create ubiquitous graphical interfaces. In *Proceedings of the 3rd international conference on Ubiquitous Computing*, 2001.
- [119] Ramesh Raskar, Jeroen van Baar, Paul Beardsley, Thomas Willwacher, Srinivas Rao, and Clifton Forlines. iLamps: geometrically aware and self-configuring projectors. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Courses*. ACM Press, 2005.
- [120] Daniel M. Russell and Rich Gossweiler. On the design of personal & communal large information scale appliances. In *UbiComp '01: Proceedings of the 3rd international conference on Ubiquitous Computing*, pages 354–361. Springer-Verlag, 2001.
- [121] Joseph F. McCarthy, Tony J. Costa, and Edy S. Liongosari. UniCast, OutCast & GroupCast: Three steps toward ubiquitous, peripheral displays. In *UbiComp '01: Proceedings of the 3rd international conference on Ubiquitous Computing*, pages 332–345. Springer-Verlag, 2001.
- [122] Keith Cheverst, Alan Dix, Daniel Fitton, Chris Kray, Mark Rouncefield, Corina Sas, George Saslis-Lagoudakis, and Jennifer G. Sheridan. Exploring Bluetooth based mobile phone interaction with the Hermes photo display. In *MobileHCI '05: Proceedings of the 7th international conference on Human computer interaction with mobile devices & services*, pages 47–54. ACM, 2005.

- [123] Jürgen Scheible and Timo Ojala. Mobilenin combining a multi-track music video, personal mobile phones and a public display into multi-user interactive entertainment. In *MULTI-MEDIA '05: Proceedings of the 13th annual ACM international conference on Multimedia*, pages 199–208. ACM Press, 2005.
- [124] Matthew Sharifi, Terry Payne, and Esther David. Public display advertising based on Bluetooth device presence. In *Mobile Interaction with the Real World (MIRW 2006) in conjunction with the 8th International Conference on Human Computer Interaction with Mobile Devices and Services*, Setembro 2006.
- [125] Olivier Liechti and Kenji Mase. iRemote: a platform for controlling public displays with mobile phones. In *Proceedings of IPSJ. Interaction 2001 Conference*, pages 35–36, 2001.
- [126] Simon Holloway. RFID in museums and galleries. *IT-DIRECTOR.COM*, pages <http://www.it-director.com/business/content.php?cid=9838>, Setembro 2007.
- [127] Jorge Cardoso. Enabling user interaction in installation art using mobile devices. *Interfaces Magazine*, 0(68):6–9, 2006.
- [128] Keith Mitchell and Nicholas J.P. Race. Oi! Capturing user attention within pervasive display environments. In *Workshop on pervasive display infrastructures, interfaces and applications - Workshop in conjunction with: The 4th International Conference on Pervasive Computing (PERVASIVE 2006)*, 2006.
- [129] Keith Mitchell, Nicholas J.P. Race, and Michael Suggitt. iCapture: Facilitating spontaneous user-interaction with pervasive displays using smart devices. In *Pervasive Mobile Interaction Devices (PERMID 2006) - Mobile Devices as Pervasive User Interfaces and Interaction Devices - Workshop in conjunction with: The 4th International Conference on Pervasive Computing (PERVASIVE 2006)*, Maio 2006.
- [130] Jakob E. Bardram, Thomas Riisgaard Hansen, Martin Mogensen, and Mads Søgaaard. Experiences from real-world deployment of context-aware technologies in a hospital environment. In *Ubicomp 2006*, pages 369–386, 2006.
- [131] Matthias Lampe and Steve Hinske. Integrating interactive learning experiences into augmented toy environments. In Anthony LaMarca, Marc Langheinrich, and Khai Truong, editors, *Pervasive Learning 2007: Design Challenges and Requirements Workshop At the*

*5th International Conference on Pervasive Computing (Pervasive2007)*, volume 4480 of *LNCS*, Berlin Heidelberg, New York, Maio 2007. Springer.

- [132] Margaret Fleck, Marcos Frid, Tim Kindberg, Eamonn O'Brien-Strain, Rakhi Rajani, and Mirjana Spasojevic. Rememberer: A tool for capturing museum visits. In *Proceedings of Ubicomp 2002*, pages 48–55, 2002.
- [133] Helder Pinto, Rui Jose, and Jose Creissac Campos. An interaction model and infrastructure for localized activities in pervasive computing environments. In *ICPS'07: IEEE International Conference on Pervasive Services*, pages 232–241, The Marmara Istanbul, Julho 2007.
- [134] Tara Matthews, Anind K. Dey, Jennifer Mankoff, Scott Carter, and Tye Rattenbury. A toolkit for managing user attention in peripheral displays. In *UIST '04: Proceedings of the 17th annual ACM symposium on User interface software and technology*, pages 247–256. ACM, 2004.
- [135] D. Scott McCrickard, C. M. Chewar, Jacob P. Somervell, and Ali Ndiwalana. A model for notification systems evaluation - assessing user goals for multitasking activity. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 10(4):312–338, 2003.
- [136] Zachary Pousman and John Stasko. A taxonomy of ambient information systems: four patterns of design. In *AVI '06: Proceedings of the working conference on Advanced visual interfaces*, pages 67–74, 2006.
- [137] Jennifer Mankoff, Anind K. Dey, Gary Hsieh, Julie Kientz, Scott Lederer, and Morgan Ames. Heuristic evaluation of ambient displays. In *CHI '03: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 169–176. ACM Press, 2003.
- [138] Donald A. Norman. *The Psychology of Everyday Things*. Basic Books, 1988.
- [139] Alan Dix, Janet E. Finlay, Gregory D. Abowd, and Russell Beale. *Human-Computer Interaction*. Prentice Hall, 3rd edition, 2003.
- [140] Alan Dix. Beyond intention - pushing boundaries with incidental interaction. In *Building Bridges: Interdisciplinary Context-Sensitive Computing*, 2002.
- [141] Erving Goffman. *Behaviour in Public Places*. The Free Press, 1966.

- [142] Paul Dourish. *Where The Action Is - The Foundations of Embodied Interaction*. MIT Press, 2001.
- [143] Elaine M. Huang and Elizabeth D. Mynatt. Semi-public displays for small, co-located groups. In *CHI '03: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 49–56, New York, NY, USA, 2003. ACM.
- [144] Michael Koch. Supporting community awareness with public shared displays. In *Proceedings of 18th Bled eConference 'Integration in Action'*, 2005.
- [145] Saul Greenberg, Michael Boyle, and Jason Laberge. PDAs and shared public displays: Making personal information public, and public information personal. *Personal and Ubiquitous Computing*, 3(1-2):54–64, Março 1999.
- [146] Karin Leichtenstern, Alexander De Luca, and Enrico Rukzio. Analysis of built-in mobile phone sensors for supporting interactions with the real world. In *Proceedings of the Pervasive Mobile Interaction Devices workshop (PERMID 2005)*, pages 31–34, 2005.
- [147] Sybase. RFID Anywhere. <http://www.sybase.com/products/rfidsoftware/rfidanywhere>, 2007 (Último acesso: Outubro 2007).
- [148] Sun. Sun Java system RFID software 3.0 developer's guide. Technical report, Sun Microsystems, Inc., 2006.
- [149] MySQL AB. MySQL DB server. <http://www.mysql.com/>, 2007 (Último aceso: Junho 2007).
- [150] JCP. JSR-82. <http://jcp.org/en/jsr/detail?id=82>, 2007 (Último acesso: Setembro 2007).
- [151] Wolfgang M. Meier. eXist - open source native XML database. <http://exist-db.org/>, 2007 (Último acesso: Junho 2007).
- [152] Sun Developer Network (SDN). Applets. <http://java.sun.com/applets/>, 2007 (Último acesso: Outubro 2007).
- [153] Sun Microsystems. Java API for XML Processing. <http://java.sun.com/j2ee/1.4/docs/tutorial/doc/JAXPIntro.html>, 2007 (Último acesso: Junho 2007).

- [154] Sun Developer Network (SDN). The Java ME device table. <http://developers.sun.com/mobility/device/device>, 2007 (Último acesso: Junho 2007).
- [155] Stefan Haustein. kXML. <http://kxml.sourceforge.net/kxml2/>, 2005 (Último acesso: Maio 2007).
- [156] Timo Salminen, Simo Hosio, and Jukka Rieki. Enhancing Bluetooth connectivity with RFID. In *PERCOM '06: Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications*, pages 36–41, Washington, DC, USA, 2006. IEEE Computer Society.
- [157] Katherine Albrecht and Liz McIntyre. *Spychips*. Nelson Current, 2005.
- [158] AJ. Solon, MJ. Callaghan, J. Harkin, and TM. McGinnity. Case study on the Bluetooth vulnerabilities in mobile devices. *International Journal of Computer Science and Network Security*, 6(4):125–129, 2006.
- [159] Rui M. Jesus, Tiago Martins, Rute Frias, Arnaldo J. Abrantes, and Nuno Correia. PhotoNav: A system to capture, share and access personal memories. In *Memories for Life Colloquium*, 2006.

## Apêndice A

# Plataforma RFID Anywhere

Este apêndice é dedicado a detalhar os componentes que integram a plataforma RFID Anywhere e o processo de simulação seguido para o protótipo divingForPearls.

O RFID Anywhere foi desenvolvido tendo por base a plataforma .NET e é executado como serviço .NET num servidor. Esta plataforma apresenta as seguintes características consideradas centrais: arquitectura distribuída; abstracção de *hardware*, protocolos e dados; opções de desenvolvimento flexíveis; gestão de rede; grande nível de segurança; e facilidade de uso. A abrangência desta solução permite ainda que a mesma possa ser considerada uma solução de *edgware* [11], ao invés de uma vulgar interface *middleware*, pois engloba diversos níveis da arquitectura de um sistema RFID.

### A.1 Componentes da Plataforma *Middleware*

Esta plataforma integra os seguintes componentes:

- **Conectores:** São uma colecção de bibliotecas responsável pela abstracção das interfaces de *hardware* específico e/ou de métodos de integração empresariais. São os conectores que recebem os dados em formato *raw* (bruto) a partir do *hardware*. Apresentam uma interface Web simples para a configuração de detalhes específicos relativos à tecnologia utilizada. O RFID Anywhere disponibiliza conectores nativos para o leitor da Alien utilizado no protótipo, simplificando a implementação.
- **Controladores:** Este componente faz a abstracção de uma norma ou protocolo específico para uma família de dispositivos de *hardware* (i.e. leitores multi-protocolo). Os controladores são independentes dos fabricantes e modelos, tendo apenas a preocupação de saber

como e para onde enviar os dados. Fazem o carregamento dos conectores e não podem ser modificados pelos programadores, mas apresentam métodos de programação que podem ser usados para o controlo do *hardware*.

- **Consola de Administração:** Esta consola é usada para a configuração e gestão do RFID Anywhere e dos seus componentes. Servidores RFID Anywhere remotos podem ser geridos a partir desta consola Web localizada num servidor RFID Anywhere central. A partir da consola podem ser executadas, por exemplo, as seguintes tarefas: iniciar, editar, reiniciar, parar e remover instâncias de serviços locais, usando para tal o gestor de serviços (ver Figura A.1); aceder a funções de monitorização e de provisionamento; adicionar e gerir sistemas na rede RFID Anywhere; editar tanto a segurança local como a segurança de domínio.

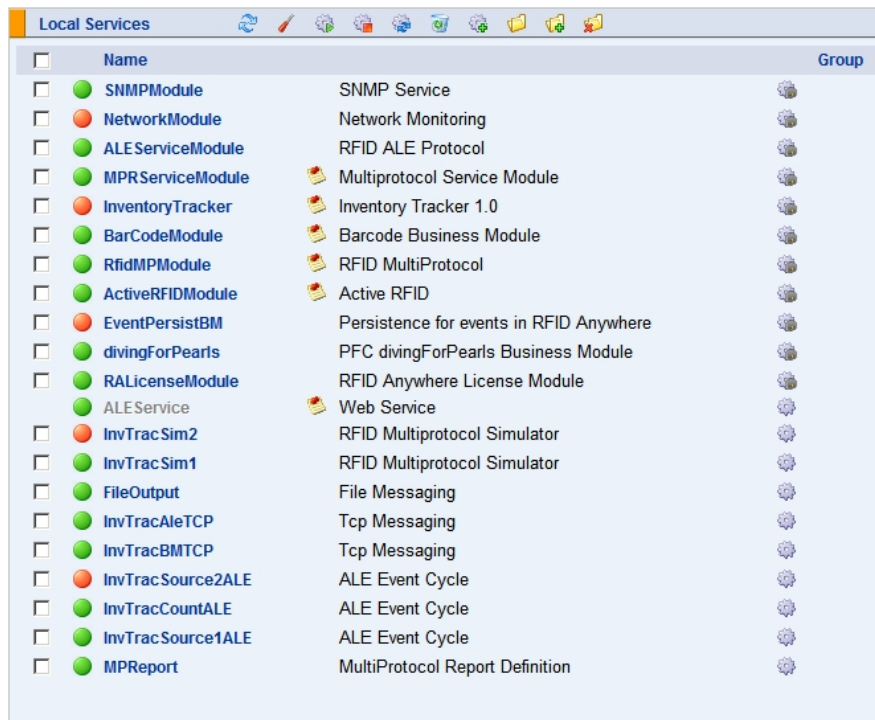


Figura A.1: RFID Anywhere: Gestor de serviços da consola de administração com serviços criados para o protótipo.

- **Enterprise Manager:** Esta é outra consola que oferece uma vista integrada de toda a rede RFID. Permite uma gestão de múltiplos nós RFID Anywhere a partir de uma localização central, o que é determinante num cenário de redes RFID multi-local, distribuídas. É instalada numa máquina dedicada e pode ser acedida de qualquer ponto na rede através de um *web browser*.

- **RFID Network Simulator:** Esta ferramenta permite que se façam simulações para cenários de teste criados (ver Figura A.2). É uma ferramenta muito útil que dispensa, numa primeira fase, a aquisição e utilização do *hardware*, permitindo que se façam testes para casos específicos ao longo do ciclo de desenvolvimento do sistema.

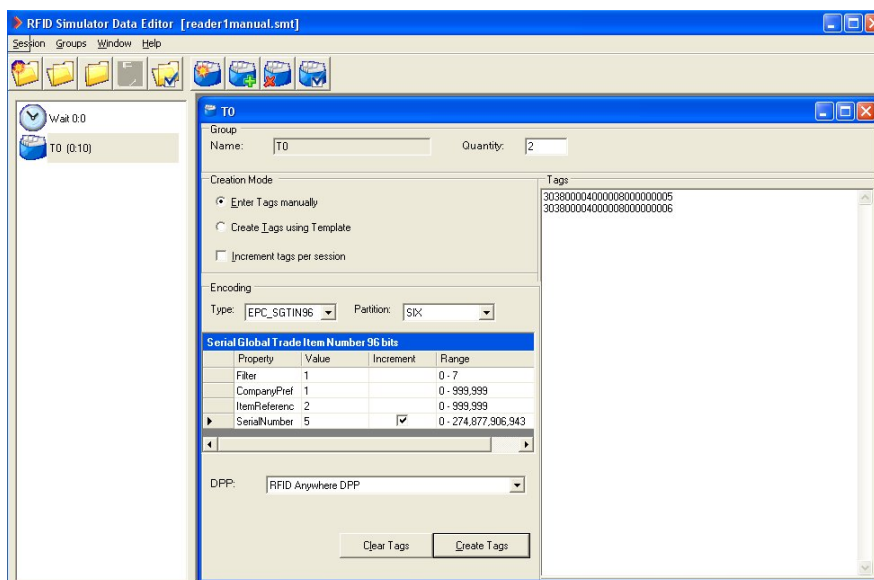


Figura A.2: RFID Anywhere: Editor de dados do RFID Simulator.

- **Location Information System (LIS):** A versão 3 do pacote já apresenta este componente que permite trabalhar com o rastreamento de itens, utilizando tecnologias tais como RFID passivo, RFID activo, *real-time location systems* (RTLS), *barcode scanners* e sensores ambientais.
- **Componentes para filtragem de dados e lógica de aplicação:**
  - **Application-Level Events (ALE):** Os ALE são responsáveis pelo processo de filtragem ao permitirem que outras aplicações possam especificar e receber apenas os dados de que necessitam. A filtragem melhora a largura de banda de rede, permite que as aplicações não tenham que processar dados não desejados, e elimina dados sem valor empresarial. Com este componente, os dados podem ser filtrados por: localização (e.g., apenas do leitor nos objectos X e Y); intervalos temporais (e.g., de 2 em 2 segundos); e tipo de item (e.g., apenas utilizadores de um dado perfil). O motor de ALE permite também que os itens sejam agrupados por categoria. A maior parte da configuração e consulta é feita pela consola de administração e os dados são transmitidos na forma de relatórios de ciclos de eventos. A interface com o ALE

pode ser feita através de programação. Permite a criação de relatórios em XML, mas encontra-se limitado ao fazê-lo apenas para *tags* EPC.

- **Report Engine MP:** Este componente permite a criação de relatórios em XML baseados na actividade de *tags* de múltiplos protocolos, e tanto para *tags* EPC como para *tags* ISO. As capacidades de processamento também permitem filtragem, agrupamento e agregação, que podem ser efectuadas de forma declarativa através da consola de administração.
- **Business Modules:** Este componente complementa os anteriores permitindo que possam ser integradas no RFID Anywhere bibliotecas (API's) criadas especificamente com a lógica da aplicação. Estes módulos oferecem aos programadores um controlo explícito sobre a filtragem sem que para tal seja necessário conhecer a interface do *hardware*. Através de *business modules*, os programadores podem preparar aspectos como a resposta a eventos de *tags*, enviar comandos de alto nível para os leitores, ou enviar informação relevante para as aplicações. São escritos em C# com a ajuda de uma extensão .NET para o Visual Studio. Com a criação de um módulo, as classes e os métodos mais usuais são automaticamente gerados, permitindo que os programadores se concentrem na criação da lógica aplicacional que os vai usar. A comunicação dos módulos pode ser feita com todos os tipos de dispositivos presentes nos controladores.
- **Camada SOA (*Service Oriented Architecture*):** A camada SOA é responsável pela disponibilização dos recursos para os outros participantes na rede como serviços independentes. O RFID Anywhere permite que múltiplas localizações possam recolher dados RFID. Os *Messaging Connectors* podem disponibilizar os dados de saída, através de relatórios XML ou de *business modules*, para múltiplos sistemas sem que seja necessário um desenvolvimento complexo. Por outro lado, também podem ser utilizados *web services*, assim como interfaces para permitir a integração com sistemas específicos.
- **Extensão para o Microsoft Visual Studio:** Esta extensão é uma ferramenta muito útil que fornece *templates* para simplificar e acelerar o desenvolvimento de aplicações RFID. Facilita a criação de *business modules* em C#, que depois são instalados, através da consola de administração, como serviços locais disponibilizados pelo *middleware*. De referir que esta extensão apenas é compatível com o Visual Studio.

## A.2 Simulação Realizada para o Protótipo divingForPearls

Utilizando o componente RFID Simulator Data Editor, o processo dividiu-se nas seguintes fases:

1. Foram criadas as *tags* que iriam aparecer junto a um leitor atribuído a um dos objectos. O ambiente de simulação agrupa as *tags* de acordo com a variável tempo, fazendo com que *tags* que são detectadas no mesmo instante e que permaneçam na área durante o mesmo tempo façam parte de um só grupo. Assim sendo, foi possível declarar quais e quantas *tags* (utilizadores/clientes) apareciam junto a um objecto em determinados instantes, tendo sido criados 10 grupos diferentes. A título de exemplo, para a criação das *tags* foi escolhida uma codificação EPC do tipo "EPC\_SGTIN96" com uma partição de "SIX". Uma tag SGTIN96 apresenta quatro campos: "*filter*", "*company prefix*", "*item reference*", e "*serial number*". Apesar de a aplicação vir a saber qual a correspondência entre *tag* e utilizador através do módulo Interpretador Contextual, aproveitou-se o "*item reference*" para atribuir a uma *tag* a identificação do seu utilizador e o "*company prefix*" para determinar o perfil. A Figura A.2 ilustra a criação de um grupo de *tags* para o qual seria, posteriormente, especificado o instante de detecção pelo leitor.
2. O passo seguinte foi a criação de um conector *RFID Multiprotocol Simulator* (InvTrac-Sim1) para simular o aparecimento das *tags* que foram definidas para simulação. Usando o conector, os dados simulados são enviados para o RFID Anywhere através do controlador *Multiprotocol Reader*. Este protocolo ficou associado ao ficheiro de simulação criado. A partir daqui poderiam ser criados e utilizados serviços *ALE*, *Report Engine* e *Business Modules* para tratar os dados simulados, tal como aconteceria se de dados reais se tratassem.
3. Por último, foram criados outros serviços como o InvTracSource1ALE, um conector ALE Event Cycle que estava configurado para filtrar e agrupar certas *tags* de um determinado perfil, e o InvTracAleTCP, que é um conector *messaging* para enviar os dados através do protocolo TCP. Este conector ficou configurado para utilizar a porta 10001, que poderá ser ouvida pela aplicação principal. Por outro lado, também foi criado um serviço para um componente *Report Engine MP* (MPReport), com as mesmas configurações de filtragem do InvTracSource1ALE, e um serviço conector *File Messaging* (FileOutput) para associação do ficheiro XML com o relatório. A Figura A.3 mostra a saída de um exemplo de relatório

## A.2. SIMULAÇÃO REALIZADA PARA O PROTÓTIPO DIVINGFORPEARLS

```
<xrnet:MPReport xrnet:xsd="http://www.w3.org/2001/XMLSchema" xrnet:xsi="http://www.w3.org/2001/XMLSchema-instance" xrnet:schemaVersion="1" xrnet:creationDate="2006-11-18T09:20:56.0000000-05:00">
  - <xrnet:report xrnet:reportName="MPReport" xrnet:reportID="91CA445D-C9F4-4CD3-85DC-9093EB403ED0" xrnet:date="2005-11-23T11:18:23.1328447-05:00"
    xrnet:totalMilliseconds="10004" xrnet:terminationCondition="DURATION">
    + <xrnet:reportDef></xrnet:reportDef>
    - <xrnet:groups>
      - <xrnet:group name="Group1">
        - <xrnet:groupDef>
          <xrnet:query>select * from sgtin96 where CompanyPrefix=1</xrnet:query>
          <xrnet:groupCount>
          - <xrnet:groupList>
            <xrnet:member id="303800004000008000000005" Uri="urn:epc:tag:sgtin-96:1.000001.000002.5" Filter="1" CompanyPrefix="1" ItemReference="2" SerialNumber="5"
              Partition="SIX" Encoding="sgtin96"/>
            <xrnet:member id="303800004000008000000006" Uri="urn:epc:tag:sgtin-96:1.000001.000002.6" Filter="1" CompanyPrefix="1" ItemReference="2" SerialNumber="6"
              Partition="SIX" Encoding="sgtin96"/>
            <xrnet:member id="303800004000020000000001" Uri="urn:epc:tag:sgtin-96:1.000001.000008.1" Filter="1" CompanyPrefix="1" ItemReference="8" SerialNumber="1"
              Partition="SIX" Encoding="sgtin96"/>
            <xrnet:member id="303800004000064000000005" Uri="urn:epc:tag:sgtin-96:1.000001.000025.5" Filter="1" CompanyPrefix="1" ItemReference="25" SerialNumber="5"
              Partition="SIX" Encoding="sgtin96"/>
            <xrnet:member id="303800004000078000000002" Uri="urn:epc:tag:sgtin-96:1.000001.000030.2" Filter="1" CompanyPrefix="1" ItemReference="30" SerialNumber="2"
              Partition="SIX" Encoding="sgtin96"/>
          </xrnet:groupList>
        </xrnet:groupDef>
      </xrnet:group>
    </xrnet:groups>
  </xrnet:report>
</xrnet:MPReport>
```

Figura A.3: divingForPearls - RFID Anywhere: Exemplo de um relatório XML gerado de um *Report Engine MP*.

XML para o serviço criado.

