Towards Pervasive Smart Spaces: A Tale of Two Projects

Telma MOTA[1], Nigel BAKER[2] , Boris MOLTCHANOV[3], Ioanna ROUSSAKI[4] , Frank KORBINIAN[5]
[1]*Potugal Telecom, Rua Eng. Pinto Basto, Aveiro, 3810-106, PORTUGAL*
Tel: +351961040913, Fax: +351234377901, Email: telma@ptinovacao.pt
[2] *CCCS Research, UWE Bristol, Frenchay, Bristol, BS16 1QY, UK*
Tel: +44 1179656261, Email: nigel.baker@uwe.ac.uk
[3]*Telecom Italia Lab, Via Olivetti. 6. Turin, 10148, ITALY*
Tel: +30 011 2287094. Fax: +39 06 418 69056, Email: boris.moltchanov@telecomitalia.it
[4]*National Technical University of Athens, 157-73 Zographou, Athens, GREECE*
Tel: +30 2107722422, Fax: +30 2107722530, Email: ioanna.roussaki@cn.ntua.gr
[5]*German Aerospace Center (DLR), Oberpfaffenhofen, 82234 Wessling, GERMANY*
Tel: +49 8153 28 2792, Fax: +49 8153 28 4 2792, Email: korbinian.frank@dlr.de

Abstract: Although PERSIST and C-CAST have different specific objectives they do share a common goal of making use of context and advancing knowledge in building pervasive smart spaces. This paper compares the approaches, analyses the results and considers the outcomes for mobile context-aware services and pervasive computing.

Keywords: context, context-awareness, smart spaces, pervasive computing.

1. Introduction

In the real world being aware of context and communicating context is a key part of human interaction. Context is a rich and powerful concept particularly for mobile users and can make network services more personalised and useful. Location and presence are examples of context based services widely deployed today. Harvesting of context to reason and learn about user behaviour will enhance the “internet of services” or “cloud computing” vision allowing services to be composed and customised according to user context. The concept of awareness and context aware applications and systems is a much more difficult proposition. Context awareness refers to the capability of an application, service or even an artefact being aware of its physical environment or situation and responding proactively and intelligently based on such awareness. Context-aware applications, context-aware artefacts or context aware systems are aware of their environment and circumstances and can respond intelligently. The ubiquity of mobile devices and proliferation of wireless networks will allow everyone permanent access to the Internet at all times and all places. The increased computational power of these devices has the potential to empower people to generate their own applications for innovative social and cognitive activities in any situation and anywhere. This wireless connection is not limited to user devices, almost any artefact from clothing to buildings can be connected and collaborate creating the “internet of things” within Web3.0 concept. Furthermore new sensor technologies and wireless sensor networks provides environmental intelligence and the capability to sense, reason and actuate. This leads to the exciting vision of the interconnection of artefacts embedded in our real environment, forming a society of “intelligent things” and “smart spaces”. This will enable all sorts of innovative interactive pervasive applications as perceived by Weiser [1]

Projects C-CAST [2] and PERSIST [3] are stepping stones along the path towards realising this vision each taking a slightly different approach and focusing on different research issues. However the notion of smart spaces and making use of context in order to deliver context-aware services is common to both. The following sections compare and contrast the objectives, conceptual models, role of context, context-awareness and results. It concludes with an analysis of those things that worked well, those that did not and finishes with agenda items for future research.

2. Objectives

2.1 – PERSIST Objectives and Conceptual Model

The objective of PERSIST is to develop Personal Smart Spaces (PSS) that provide a minimum set of functionalities which can be extended and enhanced as users encounter other smart spaces during their everyday activities. They will be capable of learning and reasoning about users, their intentions, preferences and context. They will be endowed with pro-active behaviours, which enable them to share context information with neighbouring Personal Smart Spaces, resolve conflicts between preferences of multiple users, make recommendations and act upon them, prioritise, share and balance limited resources between users, services and devices, reason about trustworthiness to protect privacy and be sufficiently fault-tolerant to guarantee their own robustness and dependability. The main challenge addressed in PERSIST is to build an intelligent pervasive computing environment capable of supporting both individual users and groups of users via the paradigm of pro-active Personal Self-Improving Smart Spaces.

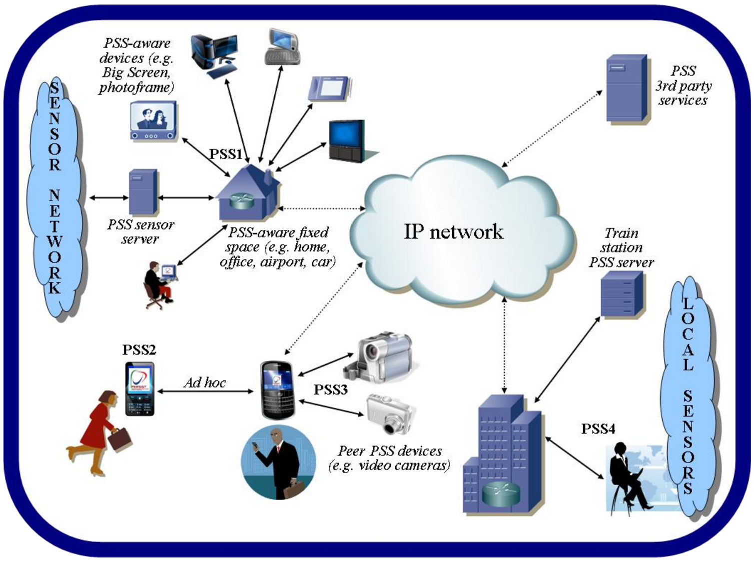

Figure 1: PERSIST Conceptual Model

2.2 – C-CAST Objectives and Conceptual Model

Project Context Casting (C-CAST) main objective is to investigate and define ways to use the situation/environment of a user (a mobile device) to initiate group communication. This environment mediated multicast may be triggered by an event or something in the physical environment offering a situation or context orientated service. Use of mobile multicast communication especially for multimedia content has always been a very attractive proposition. The motivation is derived from the fact that there will be many situations when a multicast or broadcast will be a much more effective and appropriate form of communication service however the majority are dynamic, situation specific and consequently great service opportunities are lost. C-CAST is based on two main competence areas: creation of context awareness and multicasting technologies. Context information defines groups that demand the same information or service. These services are delivered efficiently by multicasting bearers. In contrast to PERSIST the conceptual model is one of an environmental network that collects, processes context, reasons and together with models triggers appropriate multicast services as illustrated in figure 2. The C-CAST context system transforms public spaces into “personalised-group smart spaces” for content delivery. C-CAST scenarios are based on the principle that people during their daily routines often share common interests and exhibit similar behaviour especially when in public spaces such as railways stations, shopping malls or city centres. Context information gathered at any point in time is partial and incomplete but combined with environment information from sensors and with behavioural studies and models of such places enables the triggering of useful and assistive situational information. C-CAST takes advantage of a broader framework and considers as well as users, network operators, service providers and content providers.

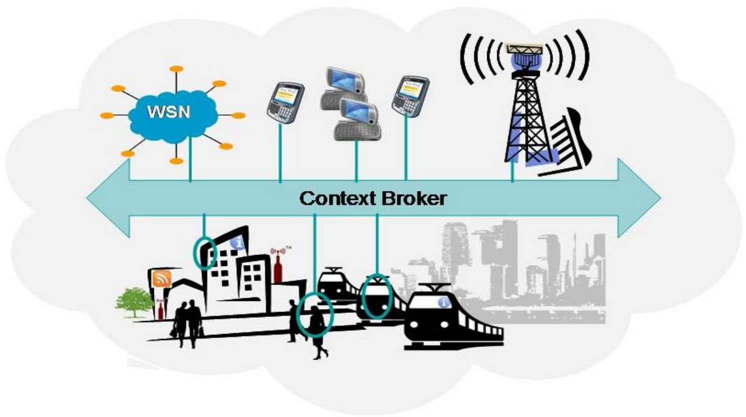

Figure 2: C-CAST Conceptual Model

3. Context Management

Context management is a key issue in designing pervasive computing systems both projects make use of a broker architecture but take slightly different approaches.

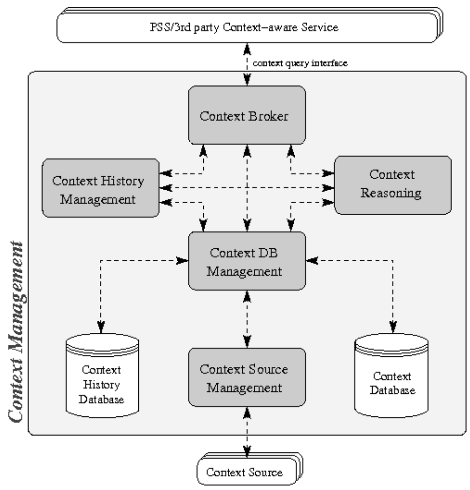

Figure 3: High-level Context Management architecture of PERSIST

3.1 – Context Management in PERSIST

As illustrated in Figure 3, the CM framework acts as an intermediate layer between PSS/3rd party context-aware services and the sources of context information. The *Context Broker* provides context consumers with a query interface for retrieving, adding, removing, and updating context data and also enables inter-communication among the CM subsystems of PSS devices. *Context Reasoning* uses various probabilistic methods in order to derive high-level context information from raw sensor data and/or context history. There are five main approaches that are offered which are: history-based context estimation, clustering to discover recurring locations, proximity estimation with a diffusion model, Bayesian filters to refine location accuracy and Bayesian high-level context inference. *Context History Management* collects, maintains and processes historic context data. It supports estimation of current and prediction of future context information based on periodic context data patterns extracted from HoC, as well as provision of the necessary training data sets to other components, thus facilitating the self-improving and pro-active behaviour of the PSS. *Context DB Management* translates context queries into standard SQL queries, which are then executed in the underlying databases. *Context Source Management* manages context sources and collects the information they monitor.

3.2 – Context Management in C-CAST

The basic context management architecture is illustrated in Figure 4 and has been specifically designed to support mobile context based services over any network. This Context Provider-Consumer model together with the Context Broker model allows the system to scale both in terms of physical distribution and in terms of context reasoning complexity. The challenge is that communicating simple context over wireless networks between constrained mobile devices requires lightweight representation whereas reasoning about situations and environments requires richer representation and models. The approach taken in this work is to define a lightweight Context Meta Language (ContextML) with which to propagate and communicate basic context. More heavyweight database models and reasoning engines also make use of this mark-up scheme to send and receive context.

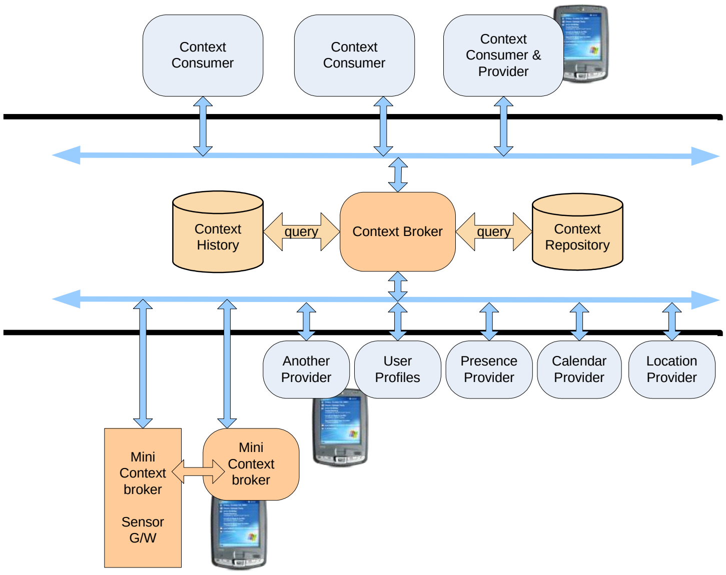

Figure 4: C-CAST Context Management Architecture

Network components take the role of Context Provider (CxP) or Context Consumer (CxC) and can exchange ContextML in HTTP messages using the Representational State Transfer (REST) interface. Consequently Context Providers and Consumers can be ported onto almost any networked device. The Context Broker (CxB) registers all CxP entities and offers a directory and lookup service so Context Consumers can find and access Providers. Registration and lookup of Context Providers is based on subjects of interests, i.e. context scopes/type (e.g. geographic location) and the entities (e.g. user, terminal) related to the contextual information. Context Providers derive context information from physical, virtual or logical sensors. Each CxP provides a specific type and scope of context information and advertises this to the Context Broker. A CxC may be any kind of application or actuator utilising the context information. Context consumers can query the CxB for a particular type of CxP and, if available, can directly query the CxP for context. This is a 'synchronous' method of communication. Another method of context query by CxC is available which is asynchronous in nature. The CxC subscribes for context information with the CxB. More details can be found in [4].

4. Context-awareness

Context awareness refers to the capability of an application, service or even an artefact being aware of its physical environment or situation and responding proactively and intelligently based on such awareness. The development of context aware applications is a complex task not only because of the need to accommodate for a wide variety of context types and their values, including the ones that cannot be anticipated at the time when the system is designed but because we are entering into the realms of autonomous systems. Consequently the approach taken is to design a flexible context infrastructure capable of learning, reasoning and adapting to different application needs.

4.1 – Personalisation, Learning and Proactivity in PERSIST

To integrate seamlessly into the users' lives context information and preferences have to be learnt and predicted from the user's daily behaviour and used to decide which services to trigger. A variety of mining and incremental machine learning methods based on the context history are used to infer and establish new and current context as illustrated in figure 5. User preferences can be learnt or manually set. A Preference Condition Monitor (PCM) applies any preference at the appropriate time. A Decision Maker component decides when to start which service proactively. With this architecture, the vision of "the right service at the right time" can become reality.

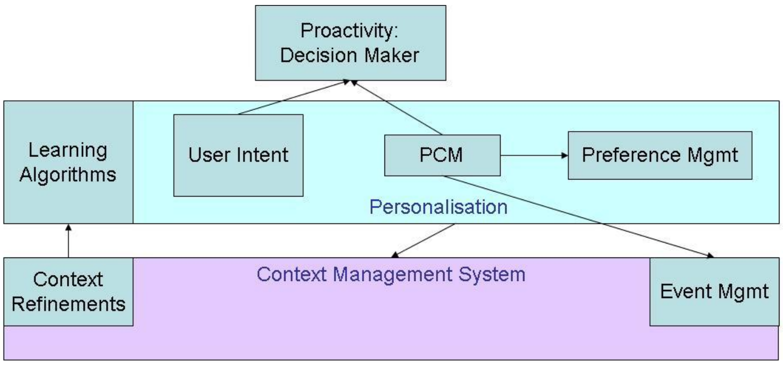

Figure 5: Persist Personalisation subsystem

4.2 – Building up Awareness in C-CAST

A particular space and hence its containing wireless network can only become aware of situations and users within it by gathering context. The model is one of many context-providers which together with personal, location and environment data increase the knowledge of users and the space in which they are situated. However to trigger a service requires recognition of a group activity or situation. This in turn requires, even for the simplest applications, some element of reasoning and learning. These more complicated computation elements are also wrapped with the context provider-consumer interface. The model is one of collaborating context providers and consumers. A Situation Provider (SP) is an example of such a more complicated component. In general a provider makes uses of several models and abstractions to complete this computational task. The main conceptual layers are illustrated in figure 5. The provider finds and subscribes to relevant context sources though the context broker.

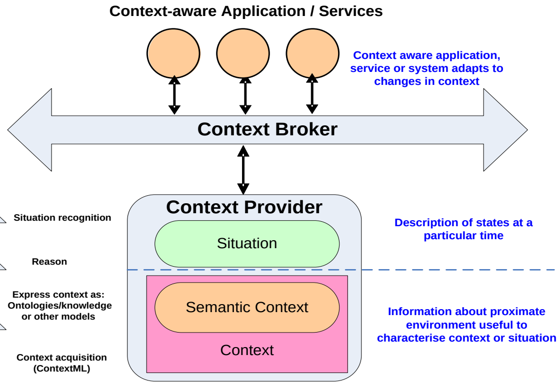

Figure 6: C-CAST Situation Context Provider

This is used to update models of the particular situations it is trained to reason about or identify. Various source-context update intervals can be defined: *rapid*, *dynamic* and *static*. These context providers run on servers within the network but conform to the same interfaces as described above.

5. Results, Conclusions and Future Work

A PERSIST smart space surrounds a user and the design concept is that of interacting personal spaces. The role of learning user preferences and user context is foremost in triggering context-aware services. Context and preferences are handled locally on many devices. This approach is good for delivering personalised services. By concentrating on the personal behaviour of one user it is possible to build up useful activity and preference profiles. Generation of these data sets is crucial for activity recognition in pervasive environments. There is the need to know how good learning and recognition algorithms are by comparison with a known training data set. So on the personal level knowledge is good and can be used for interaction with other users and groups.

In contrast C-CAST concept is of a smart place which encompasses everything inside it including users. Hence the approach is to plug into the context broker architecture many context providers that provide context pertaining to the place. Context is then used to deliver context-aware place based services. The vision is to locate brokers at places making them smart. Eventually all brokers will be federated and coverage become more complete as in wireless communication. Delivering context-aware services now becomes much more of a problem because recognising an activity or situation within the space becomes very hard.

5.1 – PERSIST

Future research plans include the development of methods for trust-based context access control in order to prevent unauthorized access to personal information. As current privacy and confidentiality protection schemes are too static and not fully suitable for the dynamic environments of PSSs, more flexible methods are necessary. To this end, access control should rely on the privacy preferences of the PSS owners that change based on their current situation. Additionally, PSS interactions with entities not a priori known will require on-the-fly access control decisions subject to the trustworthiness of these entities and the sensitivity of the data involved. The future research plans of the authors also include the design and implementation of facilities that will support context sharing in intra- and inter-PSS level, as well as proactive exchange of information on the situation, interests, resources available and the context of the PSS owners. In this respect, proactive privacy-aware and utility-based context propagation will be investigated. Furthermore, the authors plan to integrate privacy-awareness features in the history of context (HoC) handling. Context history recording and management is a powerful source of information. However, recording data with respect to the activities of persons, introduces critical privacy threats. The authors plan to investigate mechanisms for intelligent privacy-aware context history management, to be exploited both in the HoC recording, as well as the HoC forgetting phases, where the system will be able to learn and detect situations where specific user context information should be discarded or stop being recorded. Finally, the authors plan to further enhance the Quality of Context exploitation mechanisms and optimise the performance of the context learning, inference and correlation mechanism using more techniques and enabling proactive inference of context information.

5.2 – C-CAST

A first prototype of the context management framework including a query based Context Broker together with a variety of Context Providers has been implemented and trialed over a commercial cellular network and is reported in [4]. The components use REST over http communicating context through ContextML, the XML-based meta language. The results from these trials show that the lightweight context broker architecture performs well in propagating context and is capable of supporting interesting context-based services and applications. The context processing required by the applications is almost exclusively confined to finding and matching context using database queries.

As the number of context providers grow complexity of finding and accessing relevant context increases. Currently we are experimenting with federated brokers and publish-subscribe mechanisms to enhance the saleability and flexibility of context propagation. One of the main aims of the research was to establish how far we could scale this producer-consumer broker model. Would it be sufficient to support the demands of new mobile context based application and services? Could it be ported on and across a diversity of devices and networks? So far the results are encouraging.

However providing context-aware services is an order of magnitude more complex than offering context-based services. At the moment we tend to conclude that matching & selecting a service or response based on context is probably the best we can achieve. That is to say it is a best effort selection of a response (adaptation) that is likely to lead to a satisfied service user. Context-aware systems attempt to recognise and understand situations in the environment responding appropriately. It has to be accepted that in this endeavour we are attempting to construct and reason about models of the physical world based on partial, uncertain and incomplete information.

How far our concept of enhanced reasoning context providers can be taken is still an open research question. There are many challenges. For example the lightweight ContextML entity-scope representation is limited in expressing semantically rich context therefore use of other representations is necessary. Valuable information will therefore be lost in translating between representations. This assumes that machine translation is possible. As the context provider models grow so does the computation required to identify situations. The inevitable consequence is that some situations will only be identified some time after the event, of no use to mobile users who in the mean time have moved on.

One of the main lessons we have learnt in designing and deploying context-aware applications is the crucial need for social and behavioural studies. In order to help compensate for the partial incomplete context information gathered behavioural models of the region and domain in which services are to be deployed is mandatory. So for example if assisting context-aware services are to be deployed in a railway station it is necessary to study how the spaces within the building are used and the social behaviour of people who use it at different times of the day and year. In addition to this as much station and train management information must also be available. Context data sets from “Living Labs” type places would be of great benefit to the research community.

In summary delivering situation-aware services in larger scale spaces is a formidable challenge. Our applications are multimedia, so limited user damage is done when media streams are out of order, with incorrect colours delivered at the wrong time. In some cases they are still an entertaining media experience.

References

[1] Weiser, M., "The computer for the 21st century", Human-computer interaction: toward the year 2000, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1995.

[2] McBurney, S., Papadopoulou, E., Taylor, N., Williams, H., Abu-Shaaban, Y., "Comparing Two Different Architectures for Pervasive Systems from the Viewpoint of Personalisation", In Proc. eChallenges (e2009)

[3] Frank, Korbinian and Robertson, Patrick and McBurney, Sarah and Kalatzis, Nikos and Roussaki, Ioanna and Marengo, Marco (2009) “A Hybrid Preference Learning and Context Refinement Architecture”. The Society for the Study of Artificial Intelligence and the Simulation of Behaviour (SSAISB) . PERSIST Workshop on Intelligent Pervasive Environments , 6.-9. Apr. 2009 , Edinburgh.

[4] Michael Knappmeyer, Nigel Baker, Saad Liaquat and Ralf Tonjes, “A Context Provisioning Framework to Support Pervasive and Ubiquitous Applications”, 4th European Conference Smart Sensing and Context Guilford, UK 2009

[5] Moltchanov, B. et al., "Context-Aware Content Sharing and Casting", ICIN 2008, Bordeaux, France, 2008.

[6] Boris Moltchanov, Michael Knappmeyer, Omri Fuchs, Elio Paschetta, “Context Management and Reasoning for Adaptive Service Provisioning”, ICUMT2009, S. Petersburg 2009