

Contextware: Bridging Physical and Virtual Worlds

Alois Ferscha

Institut für Praktische Informatik
Universität Linz
ferscha@soft.uni-linz.ac.at

Abstract. Today a variety of terms – like Ubiquitous Computing, Pervasive Computing, Invisible Computing, Ambient Intelligence, Sentient Computing, Post-PC Computing, etc. – refers to new challenges and paradigms for the interaction among users and mobile and embedded computing devices. Fertilized by a vast quantitative growth of the Internet over the past years and a growing availability of wireless communication technologies in the wide, local and personal area, a ubiquitous use of “embedded” information technologies is evolving. Most of the services delivered through those new technologies are services adapted to *context*, particularly to the person, the time and the place of their use. The aim for seamless service provision to anyone (personalized services), at any place (location based services) and at any time (time dependent services) has brought the issues of software *framework design* and middleware to a new discussion: it is expected that context-aware services will evolve, enabled by wirelessly ad-hoc networked, autonomous special purpose computing devices (i.e. “*smart appliances*”), providing largely invisible support for tasks performed by users. It is further expected that services with explicit user input and output will be replaced by a computing landscape sensing the physical world via a huge variety of electrical, magnetic, optical, acoustic, chemical etc. sensors, and controlling it via a manifold of actuators in such a way that it becomes merged with the virtual world. Applications and services will have to be greatly based on the notion of context and knowledge, will have to cope with highly dynamic environments and changing resources, and will need to evolve towards a more implicit and proactive interaction with users.

In this paper we explore the software engineering issues, challenges and enabling technologies associated with the provision of context aware services able to:

- (i) describe, gather, transform, interpret and disseminate context information within ad-hoc, highly dynamic and frequently changing computing environments,
- (ii) dynamically discover, inspect, compose and aggregate software components in order to identify, control and extend context, as well as overcome context barriers (like time, position, user preference, etc.),
- (iii) allow for dynamic interactions among software components in a scalable fashion and satisfying special requirements such as fidelity, QoS, fault-tolerance, reliability, safety and security,
- (iv) integrate heterogeneous computing environments and devices with different functionality, ability, form factor, size and limited resources wrt. processing power, memory size, communication, I/O capabilities, etc.
- (v) support the adaptation of novel forms of sensitive, situative, non-distracting user interfaces not limited to particular modes and styles of interaction, input- output devices or service scenarios.

In an analogy to the term “middleware” – generally understood as software technologies that serve to mediate between two or more separate (and usually already existing) software components – we introduce the term “*contextware*” as the core of software technologies mediating services and the context of their use, thus bridging virtual and physical worlds.

Keywords: Software Frameworks, Context Awareness, Embedded Systems, Ubiquitous Computing, Pervasive Computing, Mobility.

1. Future Pervasive Computing Environments

The vision of a future pervasive computing landscape [Bana 00] is dominated by the ubiquity of a vast manifold of heterogeneous, small, embedded and mobile devices, the autonomy of their programmed behaviour, the dynamicity and context-awareness of services and applications they offer, the ad-hoc interoperability of services and the different modes of user interaction upon those services. This is mostly due to technological progress like the maturing of wireless networking, exciting new information processing possibilities induced by submicron IC designs, low power storage systems, smart material, and motor-, controller-, sensor- and actuator technologies, envisioning a future computing service scenario in which almost every object in our everyday environment will be equipped with embedded processors [Arno 99], wireless communication facilities and embedded software to percept, perform and control a multitude of tasks and functions. Since many of these objects will be able to communicate and interact with each other, the vision of “context aware” appliances [ELMM 99] and spaces [MIT] [Essa 99] [RoCa 00] – where dynamically configured systems of mobile entities by exploiting the available infrastructure and processing power of the environment – appears close to reality. We can hypothesize that the individual utility of mobile communication, wireless appliances and the respective mobile Internet services will be greatly increased if they were *personalized*, i.e. user centered and dynamically adapted to user preference, *location aware*, i.e. multimodal and multifunctional with respect to the environment [AAHL 97], and *time dependent*, i.e. if they were time dynamic and exhibited timely responsiveness (the more recent literature refers to time, space and personal preference as dimensions of a *context*, and *context awareness* as the ability to respect those – and other dimensions – when delivering service to the user [DeAb 99] [Dey 01]). One impacting trend hence will be the evolution of context aware digital environments – often referred to as “smart appliances” or “smart spaces” – that intelligently monitor the objects of a real world (including people), and interact with them in a pro-active, autonomous, sovereign, responsible and user-authorized way. People will be empowered through an environment that is aware of their presence, sensitive, adaptive [Moze 99] and responsive to their needs, habits and emotions, as well as ubiquitously accessible via natural interaction [SATT 99].

The next generation of “pervasive” computing environments hence will be characterised by the following basic elements: (i) *ubiquitous access*, (ii) *context awareness*, (iii) *intelligence*, and (iv) *natural interaction*. Ubiquitous access refers to a situation in which users are surrounded by a multitude of interconnected embedded

systems, which are mostly invisible and weaved into the background of the surrounding, like furniture, clothing, rooms, etc., and all of them able to sense the setting and state of physical world objects via a multitude of sensors. Sensors, as the key enablers for implicit input from a “physical world” into a “virtual world”, will be operated in a time-driven or event-driven way, and actuators, as the generic means for implicit output from the “virtual” to the “physical world”, will respond to the surrounding in either a reactive or proactive fashion (see Figure 1). The way how humans interact with the computing environment will thus fundamentally change towards implicit, ubiquitous access. Context awareness [RCDD 98] refers to the ability of the system to recognise and localise objects as well as people and their intentions. Intelligence refers to the fact that the digital surrounding is able to adapt itself to the people that live (or artefacts that reside) in it, learn from their behaviour, and possibly recognise as well as show emotion. Natural interaction finally refers to advanced modalities like natural speech- and gesture recognition, as well as speech-synthesis which will allow a much more human-like communication with the digital environment than is possible today.

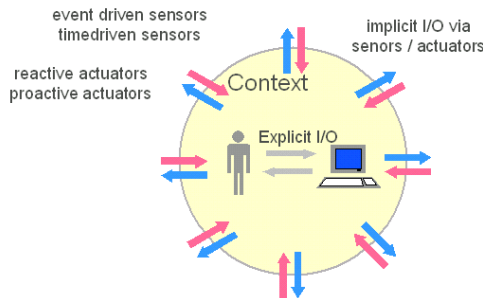


Fig. 1. Contextware: supporting implicit Sensor-Actuator controlled interactions

It appears obvious that the notion of “context” is what will be the driving force for future applications and services using global networks (like the Internet) as backbone communication infrastructure. The context of an application is understood as “any information that can be used to characterize the situation of an entity”, an entity being “a person, place or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves” [Dey 01]. A key architecture design principle for context-aware applications (see e.g. [AAHL 99]) will be to decouple mechanism for collecting or sensing context information [SDOA 99] and its interpretation, from the provision and exploitation of this information to build and run context-aware applications [Schi 95]. To support building context-aware applications, software developers should not be concerned with how, when and where context information is sensed. Sensing context must happen in an application independent way, and context representation must be generic for all possible applications [FBN 01].

2. Technology Trends and Contextware Challenges

What was envisioned as the future “pervasive” computing landscape in the previous section follows from technology trend observations in the hardware, software and communications area [HIR 01]. The challenges posed by those trends towards the realization of pervasive computing environments concern a ubiquitous communication infrastructure for devices, the support for user interaction and user mobility, the provision of universal interfaces and the adaptation to individual user needs. One of the most important concerns, however, is to design and develop software components that facilitate ad-hoc networking applications characterized by highly dynamic changes in the environment and the access modes or patterns, and their scalability with respect to the complexity of interactions and the number of involved entities or actors – i.e. the establishment of profound software methods and development processes for contextware. Towards this end, a more integrated view of software design has to be adopted, also considering hardware, communication and access trends.

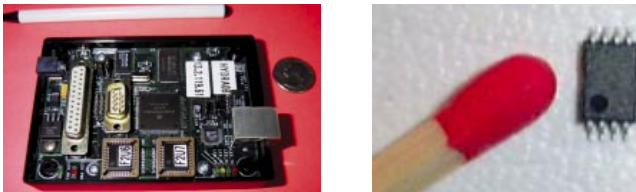
Ubiquitous access in an embedded pervasive computing environment is promisingly implemented based on a continuous, wireless communication infrastructure involving broadband satellite systems, radio communication (e.g. GSM, GPRS, TETRA, DECT, EDGE, UMTS/IMT2000, G4, Bluetooth, HomeRF, IEEE802.11, HiperLAN, HomeCast, etc.), infrared (IrDA) and ultrasonic communication, wireless sensor networks, power line communications, wireline communications and standards (USB, IEEE1394, HomePNA), etc. The primary “software” challenge here lies in the maintenance of seamless connections as devices move between different areas of different network technology and network connectivity, as well as the handling of disconnects. While communication problems like routing and handover can be handled at the network level, others cannot be solved at his level as they relate to the interaction semantics at the application level [Else 99] [Arno 99]. Device heterogeneity and wide differences in hardware and software capabilities requires a communication infrastructure that maintains knowledge about device characteristics and manages coherent device interactions (e.g. among wearable devices, home appliances and outdoor appliances). Personal area network (PAN) technologies are supposed to address these issues.

The dynamic interconnection of devices and discovery services [McGr 00] within PANs is approached with network technologies like MobileIP, IPv6, and coordination software systems [OZKT 01] like HAVI [LGDE 00], Java/Jini [Wald 99], JXTA [WDKF 02][Gong 02], Java wireless data, Java-Spaces, the Simple Object Access Protocol [BEKL], Web-OS, XML-RPC, UPnP [ChLa 99], Salutation [Mill 99], Tspaces, etc. While service registration and discovery, lookup services, self organization, caching and differencing methods have working solutions today (see Table 1 for a comparison), context based networking [Esls 99] and the context based coordination of entities and activities must still be considered as research issues.

Table 1. A Comparison of Approaches for “Coordination”

	Jini/Java	UPnP	Salutation	Tspaces
Originator	SUN Microsystems	primarily Microsoft	Industry / Academic consortium	IBM
Register / Announcing Presence:	Unicast/Multicast to Jini lookup service Lease expiry for up to date information	SSDP Protocol, working with or without Proxy Service Unicast/Multicast HTTP - UDP	through a Salutation Manager (SM) works as a service broker	making services available on a Tuple Space
Discovering other devices	Querying lookup services Optionally using an RMI proxy	SSDP (Simple Service Discovery Protocol) used to announce their presence as well as find other devices	queries on SM	Database indexing and query on the TupleSpace
Describing capabilities	Registration Information can have attribute pairs RMI interface with known methods	XML documentation is made available URIs are used	functional units which know features of a service	Java RMI Method invocation interfaces with known methods
Self configuration	not considered	DHCP or AutoIP multicast DNS	not considered, difficult because of transport independence	not considered
Invoking services:	Through the lack of device driver difficult, device must be Java enabled, RMI	Invoking Services via Browser, or direct API calls through applications	flexible, vendor specific, realised through SM	Java RMI framework
Security:	Java Security	not finished yet	allows user authentication	access controls by User
Transports:	TCP/IP – Proxies	TCP/IP – Proxies	transport independent	TCP/IP
Supported Devices	all kind of devices, mobile, household equipment	all kind of devices, mobile, household equipment	printer, FAX, telephone, PDA , computer (of any size) but not limited to them	Printers, PDAs, Computers
Remarks	no self configuration makes it weak RMI Method invocation is difficult to handle + Security	self configuration is considered as the strongest feature of this technology	advantages in the functionality of the SMS, problems with transport independence	enhancement of Linda TupleSpace concept

As far as a “pervasive Internet” is concerned, a major enabling technology is the miniaturization of devices able to accept and to respond to HTTP requests, i.e. the provision of WWW servers to be weaved into arbitrary fabrics of daily use objects. Figure 2 demonstrates popular embedded webserver miniatures (as they are readily available from e.g. HYDRA, Dallas Semiconductors, Tini, Boolean, etc.), envisioning the technological options for e.g. embedded Internet appliances.

**Fig. 2.** Miniaturized Embedded and On-Chip Webservers

New output technologies like light emitting polymers (see Figure 3, right), foldable displays, laser diodes, electronic ink, electronic paper etc. together with new materials able to serve as sensors or input technologies for monitoring the environment (digital and analog), appliances like T-shirt computers, finger rings, hot badges, smart rooms and smart environments, interactive workspaces like smart walls,

tables, chairs etc., spawn a whole new space for information appliance designs. Implicit input and output in an embedded Internet will generally be realized via sensor and actuator technologies, involving optical and opto-electrical (photodiodes, -conductors, charge coupled devices, light-emitting diodes and polymers, lasers, liquid crystals), acoustic (e.g. acoustic transducers, sonar transducers, surfaces acoustic wave devices), mechanical (metallic, silicon pressure sensors and accelerometers, solid-state displacement transducers, piezoelectric field-effect transducers), thermal (silicon transistor thermometers, diode temperature sensors, pyroelectric and quartz thermometers, integrated temperature transducers), magnetic (magnetoresistors, silicon depletion-layer magnetometers, magneto-injection transistors, MOS magnetic field sensors), chemical sensors, microwave sensors, environmental sensors, etc. A consequence of this multisensory input possibilities is the need for a “continuously present” interface, e.g. continually-worn sensors in a context-aware computing application [SSP 98], the integration of voice-vision-text technologies, gesture, eye- and body-movement recognition etc. (Figure 3 illustrates technologies enabling a “continuous” interface to e.g. an embedded Internet)

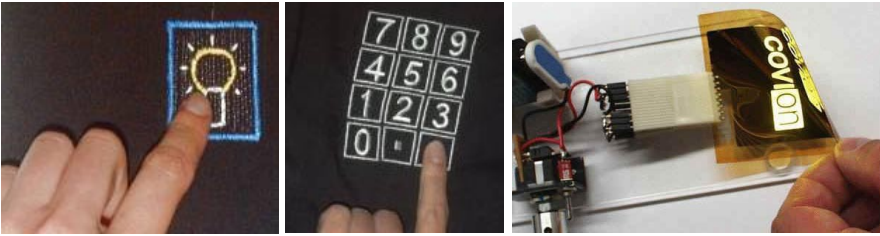


Fig. 3. New Materials for a “continuously Present” Interface (Pyroelectric material (TWO LEFT) , LEP Display (RIGHT))

3. Contextware: Bridging Physical and Virtual Worlds

To build context aware applications [SDA 99], the adoption of a world model (in the sequel referred to as “virtual world”) representing a set of objects and their state in a physical (or “real”) world (or at least the subworld essential for the specific application) is the common software engineering approach suggested in the literature [DSA 99] [DSAF 99a] [DSAF 99b]. What makes an application context aware is the ability to interact with objects in the real world, requiring adequate models and representations of the objects of interest, and the possibility to sense, track, manipulate and trigger the real objects from within the world model. Several frameworks for such world models have appeared recently [PRM 99] [KBMB 00] [KHCR 00] [FCE], the most prominent ones [HP] identifying *persons*, *things* and *places* as the primary abstract classes for real world objects. People living in the real world, acting, perceiving and interacting with objects in their environment are represented in a “virtual world” by “virtual objects” or “proxies”. Proxies of persons, things and places are linked to each other in the virtual world, such that this “linkage” is highly correlated with the “linkage” of physical persons, things and places in the real world. A context-aware application [BBC 97] now monitors the state and

activity of the real world objects via *set of sensors*, *coordinates* the proxies according to the rules embodied in the application, and notifies, triggers [Brow 98] or modifies the physical world objects via a *set of actuators*.

We claim that a context-aware application can only be “intelligent” about an object if it can *identify*, *localize* and track, and *coordinate* with respect to and relative to the other objects around:

- (i) Identification (sensing the identity of or recognizing a real world object),
- (ii) localization (sensing its position and possibly its movement in space), and
- (iii) coordination (relating it semantically to other objects and behavioral rules)

hence are the central issues for embedded pervasive computing applications, and, moreover, are essential to all context-aware applications. They are among the few aspects that are fundamentally different from conventional middleware.

A plenty of ready-to-use technologies for the automated recognition (identification) of real world objects can be accounted: technologies based on optical (barcode and OCR), magnetic (SmartCard), ultrasonic (Active Badge and iButton) sensors, voice and vision based systems, biometrical systems (fingerprint, retina, face recognition), etc.. Many of those are also suitable for short distance positioning and tracking (localization), and are already in use for locator services in many different fields of application. Global positioning technologies based on GSM, GPS, dGPS extend the range of options for long distance localization.

An identification and localization technology with a certain appeal for embedded pervasive computing applications is radio frequency identification (RFID) which is based on radio or electromagnetic propagation, i.e. the ability to allow energy to penetrate certain physical objects and read a tag that is not necessarily visible. Objects can be identified remotely, either in the form of reading an identity code, or more simply just checking for the presence of a tag. An RFID system consists of a tag (or transponder), and a reader device. The transponder as a passive component responds by replying to an interrogation request received from an interrogator. The reader as an active component induces an interrogation request, and receives back some data from the transponder (such as an identity code or the value of a measurement) with virtually zero time delay. Different frequencies of the radio system of the reader result in different reading ranges (10 cm to 1 m) and properties of the system. Commonly available tags have an operating frequency in the range from 60 kHz to 5.8 GHz, 125 kHz and 13,56 MHz being the most prominent frequencies in use today. The reason why transponders have recently started to become major players in the field of electronic identification is their contact-less and powerless operation (no power supply needed in the tag), low cost packaging, “unlimited” lifetime, ISO standard compliance (14443 A/B and 15693-2), a wide choice of qualified packaging (Smart Cards, tags, inlets, smart labels, etc.), short range operation, proximity and vicinity communication with one and the same technology, cryptographic security (i.e. the protection against unauthorised product copies or data modification), and, last but not at least, low investment level for contact-less technology integration.

Coordination [OZKT 01] finally, is the concern of software frameworks providing concerted but autonomous components for object representations of the physical world, such as persons, things and places. Coordination, in its very early understanding addresses to express the interaction among individual, abstract “agents” (humans, processes, services, devices, software objects, etc.) within a coordination model, the linguistic embodiment of which is a coordination language. Such languages have been designed to allow for a description of the dynamic creation and destruction of agents, the flows of control and communication among agents, the control of spatial distribution of agents, and the synchronization and distribution of agent actions over time. Metadata models like the RDF (Resource Description Framework) [RDF 99] for the definition and description of abstract object classes and the operations required by a applications have been successfully applied in context-aware embedded Internet applications [FBN 01]. Generally, such frameworks provide an abstract specification (or a model) of the set of relevant objects and the semantics of the operations to be supported by the application. Concrete implementations by exploiting the mechanisms of inheritance and reuse realize the (i) creation and management of virtual representations of physical objects, (ii) mechanisms for sensing (including identification, localization and tracking) and manipulating such objects, and (iii) mechanisms for the coordinated invocation of object methods.

4. A Context-aware Application Scenario

To illustrate the co-influence of hardware and software engineering issues by an example, we have developed a person/thing/place scenario and have implemented a “context-aware luggage” framework. The framework “models” the physical world as instances of the abstract object classes person, thing and place, and the contextual interrelatedness among those object instances by a set of (bilateral) object relations. Object instances can be created and interrelated at run-time, relations can be updated according to dynamic changes of context (like time, location, etc.) via time-triggered or event-triggered sensors. Figure 4 exemplifies a simple physical world scenario containing a person “ferscha” presently located in town “vienna” as the “owner” of a “suitcase”. Upon change of context, like the movement to a different place, inheritance mechanism resolve transitivity of object relations like the “is_in” relation for objects “contained” in a (geographically) moving (physical world) object.

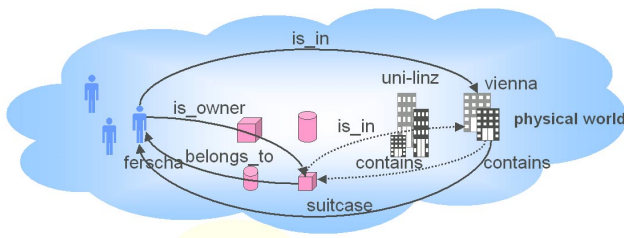


Fig. 4. Objects and their Interrelationship in a Physical World Scenario

Technologically, an embedded personal computer has been integrated into an off-the-shelf suitcase (see Figure 5), executing a standard HTTP services on top of a TCP/IP stack over an integrated IEEE802.11b WLAN adaptor. A miniaturized RFID reader is connected to the serial port of the server machine, an RFID antenna is integrated in the frame of the suitcase so as to enable the server to sense RFID tags contained in the suitcase. A vast of 125KHz magnetic coupled transponders are used to tag real world objects (like shirts, keys, PDAs or even printed paper) to be potentially carried (and sensed) by the suitcase. The suitcase itself is tagged and possibly sensed by readers integrated into home furniture, car or airplane trunks, conveyor belts etc. so as to allow for an identification and localization at any meaningful point in space of the application.

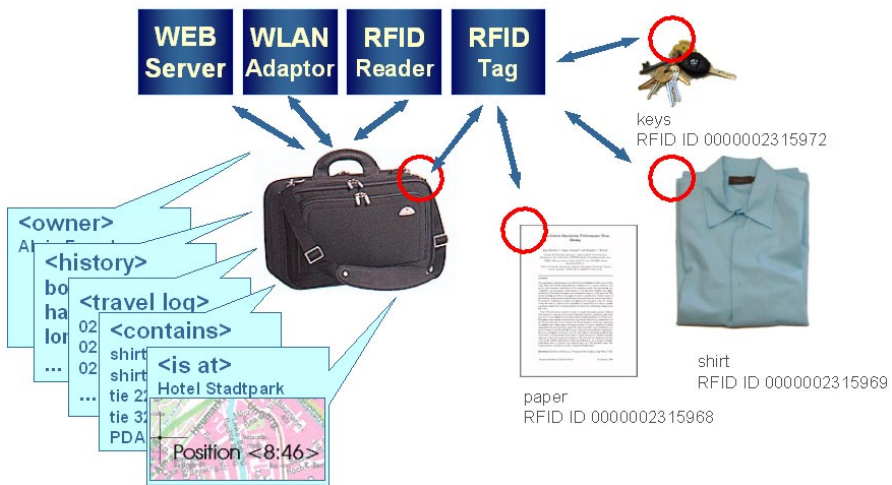


Fig. 5. “Context-Aware Luggage” as an Application instance of the Context Framework

In our context framework, an application specific abstraction of the real world is generated from three generic classes for persons, things, and places. The reification of physical world objects and their relation among each other is expressed in RDF (see Figure 6). The person object represents the concepts about a person that are necessary to link their physical properties and activities to the virtual world. The thing object encapsulates the basic abstraction for objects relevant to the application and has instances like shirt, bag, etc. The place object is used to represent the essential features of a physical location like an office room or a waiting lounge. Places, things and persons may be related in a manifold of ways.

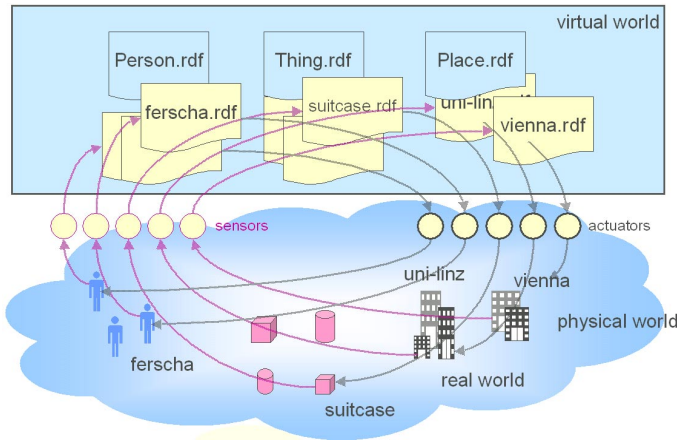


Fig. 6. RDF Model for Virtual World Representation under Sensor/Actuator Control

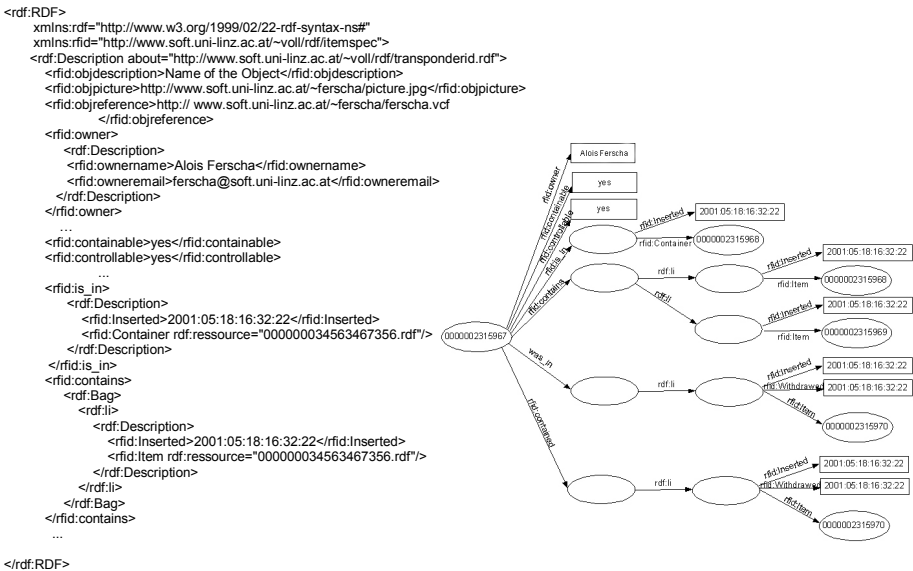


Fig. 7. Relating Physical World Objects with their Virtual Reifications

The RDF in Figure 7 sketches quite a few relations of our context framework: The *owner* relation expresses ownership of a real world object by another, the *contains* and *is_in* relations expresses geometrical containment of an object within another one, the *contained* and *was_in* relations trace the history of containment into RDF bags, the *containable* attribute defines whether an object may contain another one, the *controllable* attribute allows to prevent or enable the modification of an object RDF by the object itself, etc. The unique ID associated with every real world object is the ID encoded in its RFID tag. It is sensed by the RFID reader which triggers a script to

update the involved object RDFs (Inserting e.g. the shirt into the suitcase would cause the RFID reader to identify the shirt tag and update (among others) both the shirts RDF relation *is_in*, as well as the suitcases RDF relation *contains* by cross-referring URIs. Figure 7 left presents a snapshot of an RDF of the suitcase after some insert operations.

5. Conclusions

Recent advances in hardware-, communication- and sensor-/actuator technologies envision a whole new area of a situation dependent computing landscape, challenging the classical software design and development principles. Besides the demand for new middleware paradigms and concepts that encompass – besides software issues – also devices, users, interaction, and dynamicity of context, have created new requirements for the **automated** configuration and operation of pervasive computing applications. Central to those is the notion of “**context**”, preliminarily defined [DeAb 99][Dey 01] as any information that can be used to characterize the state or situation of an entity and the spectrum of its behaviours. Software architectures and frameworks for context-aware applications [UKC] thus must be concerned with (i) abstract representations of context, (ii) sensing, collecting, interpreting and transforming of context information and (iii) disseminating and making appropriate use of context information to impact the behaviour of the application. The methodological approach today is to employ standardized Web-metadata modelling (like SGML, XML, RDF) to solve (i) [CaDe 00], integrated electrical, chemical, magnetic, optical, acoustic etc. multi-sensor systems for (ii), and integrated multi-actuator systems involving data processors, controllers, motors, filters, etc. for (iii). Given today’s global networks as the backbone communication infrastructure, ubiquitous access to this infrastructure demands technological solutions for the “spontaneous” discovery and configuration of devices and services, the selection of meaningful services offered by autonomous software components, the automatic adaptation to mobile and sporadic availability, the interoperability across manufacturers, platforms and services and the scalability with respect to the number of involved activities and entities. On top of **ubiquitous access**, **context-awareness** and issues of **intelligence** (learning, memorizing, planning, forgetting) and knowledge processing are essential for the provision next generation context-aware applications. A very important aspect is the way how humans **interact** with those applications: requirements towards pro-active (rather than reactive) responsiveness, towards implicit (rather than explicit) input-output and towards continuously (in time and space) available, “naturally occurring” interfaces [CCB 00] will drive their development. The “linking” of real world contexts with their virtual counterparts poses three fundamental problems to be solved in a “co-investigation” of the spectrum of options in hardware for sensors and actuators, and software framework designs are related to (i) *identification* – the automated recognition of the identity of entities – (ii) *localization* and tracking of those entities as they change state in space and time, and (iii) *coordination* – i.e. to bring the autonomous activities of virtual entities to a concerted action that represents the application.

References

- [AAHL 97] G. D. ABOWD, C.G. ATKESON, J. HONG, S. LONG, R. KOOPER, M. PINKERTON: Cyberguide: A mobile context-aware tour guide. *ACM Wireless Networks*, Vol. 3, no. 5, pp. 421-433, 1997. <http://www.acm.org/pubs/articles/journals/wireless/1997-3-5/p421-abowd/p421-abowd.pdf>
- [Arno 99] D. ARNOLD ET.AL.: Discourse with disposable Computers: How and why you will talk to your tomatoes. *USENIX Workshop on Embedded Systems*, 1999.
- [Bana 00] Banavar, G. et al., "Challenges: An Application Model for Pervasive Computing", *Proceedings 6th Annual Intl. Conference on Mobile Computing and Networking (MobiCom 2000)*, August 2000.
- [Brow 98] P. J. BROWN: Triggering Information by Context. *Personal Technologies*, Vol. 2, Nr.1, pp. 18-27, 1998. <http://www.cs.ukc.ac.uk/pubs/1998/591/content.html>
- [BBC 97] P. J. BROWN, J. D. BOVEY, X. CHEN: Context-Aware Applications: From the Laboratory to the Marketplace. *IEEE Personal Communications*, Vol. 4, Nr. 5, pp. 58-64, 1997.
- [BEKL] Box, D., Ehnebuske, D., Kakivaya, G., Layman, A., Mendelsohn, N., Frystyk Nielsen, H., Thatte, S. Winer, D. SOAP: Simple Object Access Protocol. <http://www.w3.org/TR/SOAP/>.
- [CaDe 00] D. CASWELL, P. DEBATY: Creating a Web Representation for Places. *Proceedings of the Ninth International World Wide Web Conference*, 2000. <http://cooltown.hp.com/papers/placeman/PlacesHUC2000.pdf>
- [ChLa 99] B. CHRISTENSSON, O. LARSSON: Universal Plug and Play Connects Smart Devices. *WinHEC 99*, 1999. <http://www.axis.com/products/documentation/UPnP.doc>
- [CCB 00] J. L. CROWLEY, J. COUTAZ, F. BERARD: Perceptual user interfaces: Things That See. *Communications of the ACM*, Vol. 43, Nr. 3, pp. 54-64, 2000.
- [CZHI 99] S. E. CZERWINSKI, B.Y. ZHAO, T.D. HODES, A. D. JOSEPH, R. H. KATZ: An Architecture for a Secure Service Discovery Service. *Mobicom '99*, 1999. <http://ninja.cs.berkeley.edu/dist/papers/sds-mobicom.pdf>
- [DeAb 99] A. K. DEY, G. D. ABOWD: Toward a Better Understanding of Context and Context-Awareness. GIT, GVU Technical Report GIT-GVU-99-22, June 1999. <ftp://ftp.gvu.gatech.edu/pub/gvu/tr/1999/99-22.pdf>
- [Dey 01] A. K. Dey: Understanding and Using Context. *Personal and Ubiquitous Computing, Special Issue on Situated Interaction and Ubiquitous Computing*, 5(1), 2001.
- [DSA 99] A. K. DEY, D. SALBER, G. D. ABOWD: A Context-Based Infrastructure for Smart Environments. *The First International Workshop on Managing Interactions in Smart Environments (MANSE '99)*, Dublin Ireland, 1999. <http://www.cc.gatech.edu/fce/ctk/pubs/MANSE99.pdf>
- [DSAF 99a] A. K. DEY, D. SALBER, G. D. ABOWD, M. FUTAKAWA: The Conference Assistant: Combining Context-Awareness with Wearable Computing. *The Third International Symposium on Wearable Computers (ISWC '99)*, 1999. <http://www.cc.gatech.edu/fce/ctk/pubs/ISWC99.pdf>
- [DSAF 99b] A. K. DEY, D. SALBER, G. D. ABOWD, M. FUTAKAWA: An Architecture to Support Context-Aware Applications. GVU, Technical Report GIT-GVU-99-23, June 1999. <ftp://ftp.gvu.gatech.edu/pub/gvu/tr/99-23.pdf>
- [ELMM 99] K. F. EUSTICE, T. J. LEHMAN, A. MORALES, M. C. MUNSON, S. EDLUND, M. GUILLEN: A universal information appliance. *IBM Systems Journal*, Vol. 38, Nr. 4, 1999. <http://www.research.ibm.com/journal/sj/384/eustice.html>
- [Esle 99] M. ESLER ET.AL.: Next Century Challenges: Data Centric Networking for Invisible Computing. *Proceedings of the 5th Annual International Conference on Mobile Computing and Networking (MobiCom '99)*, August 1999.

- [Essa 99] I. ESSA: Ubiquitous Sensing for Smart and Aware Environments: Technologies toward the building of an Aware House. *DARPA/NSF/NIST Workshop on Smart Environments*, 1999.
<http://www.cc.gatech.edu/gvu/perception/pubs/se99/pp.pdf>
- [FCE] Future Computing Environments: The Context Toolkit. <http://www.cc.gatech.edu/fce/contexttoolkit/>
- [FBN 01] A. FERSCHA, W. BEER, W. NARZT: Location Awareness in Community Wireless LANs. *Proceedings of the Informatik 2001: Workshop on Mobile internet based services and information logistics, September 2001*.
- [Gong 02] L. GONG: Peer-to-Peer Networks in Action. *IEEE Internet Computing*, Vol. 6, Nr. 1, pp. 36-39, Jan/Feb 2002.
- [HaHo 94] A. HARTER, A. HOPPER: A distributed location system for the active office. *IEEE Network*, Vol. 8, Nr. 1, pp. 62-70, 1994.
- [HIR 01] K. HENRICKSEN, J. INDULSKA, A. RAKOTONIRAINY: Infrastructure for Pervasive Computing: Challenges. *Proc. of the Informatik 2001: Workshop on mobile internet based services and information logistics, September 2001*.
- [HP] Hewlett Packard: CoolTown Appliance Computing: White Papers. <http://cooltown.hp.com/papers.htm>
- [KBMB 00] T. KINDBERG, J. BARTON, J. MORGAN, G. BECKER, I. BEDNER, D. CASWELL, P. DEBATY, G. GOPAL, M. FRID, V. KRISHNAN, H. MORRIS, J. SCETTINO, B. SERRA, M. SPASOJEVIC: People, Places, Things: Web Presence for the Real World. *WWW'2000*, 2000. <http://cooltown.hp.com/dev/wpapers/WebPres/WebPresence.asp>
- [KHCR 00] F. KON, C. HESS, CHRISTOPHER, M. ROMAN, R. H. CAMPBELL, M. D. MICKUNAS: A Flexible, Interoperable Framework for Active Spaces. *OOPSLA'2000 Workshop on Pervasive Computing*, Minneapolis, MN, 2000.
- [LGDE 00] Lea, R., Gibbs, S., Dara-Abrams, A. and Eytchison, E., "Networking Home Entertainment Devices with HAVi", *Computer*, Vol. 33, No. 9, September 2000.
- [McGr 00] R. E. MCGRATH: Discovery and its Discontents: Discovery Protocols for Ubiquitous Computing. Department of Computer Science University of Illinois Urbana-Champaign, Urbana UIUCDCS-R-99-2132, March 25, 2000. http://www.cs.uiuc.edu/Dienst/Repository/2.0/Body/ncstrl.uiuc_cs/UIUCDCS-R-2000-2154/pdf
- [Mill 99] B. MILLER: Mapping Salutation Architecture APIs to Bluetooth Service Discovery Layer. Bluetooth Consortium I.C.118/1.0, 01 July 1999.
- [MIT] MIT Media Lab: Smart Rooms. <http://ali.www.media.mit.edu/vismod/demos/smartroom/>
- [Moze 99] M. C. MOZER: An intelligent environment must be adaptive. *IEEE Intelligent Systems and Their Applications*, Vol. 14, Nr. 2, pp. 11-13, 1999. <http://www.cs.colorado.edu/~mozer/papers/ieee.html>
- [OZKT 01] A. OMICINI, F. ZAMBONELLI, M. KLUSCH, R. TOLKSDORF (EDS.): Coordination of Internet Agents. Models, Technologies, and Applications. *Springer Verlag*, Berlin 2001.
- [PRM 99] J. PASCOE, N. RYAN, D. MORSE: Issues in developing context-aware computing. In H-W.Gellersen, editor, *Handheld and Ubiquitous Computing*, Nr. 1707 in Lecture Notes in Computer Science, pages 208-221, Heidelberg, Germany, September 1999. Springer-Verlag
- [RDF 99] O. LASSILA, R. R. SWICK: Resource Description Framework (RDF): Model and Syntax Specification. Recommendation, World Wide Web Consortium, Feb. 1999. <http://www.w3c.org/TR/REC-rdf-syntax/>.
- [RoCa 00] M. ROMAN, R. H. CAMPBELL: GAIA: Enabling Active Spaces. In *9th ACM SIGOPS European Workshop*. September 17th-20th, 2000. Kolding, Denmark. <http://choices.cs.uiuc.edu/2k/papers/sigopseu2000.pdf>
- [SaAb 98] D. SALBER, G. D. ABOWD: The Design and Use of a Generic Context Server. *Perceptual User Interfaces Workshop (PUI '99)*, San Francisco, 1998. <http://www.cs.gatech.edu/fce/contexttoolkit/pubs/pui98.pdf>

- [SDA 99] D. SALBER, A. K. DEY, G. D. ABOWD: The Context Toolkit: Aiding the Development of Context-Enabled Applications. *The 1999 Conference on Human Factors in Computing Systems (CHI '99)*, Pittsburgh, 1999. <http://www.cs.gatech.edu/fce/contexttoolkit/pubs/chi99.pdf>
- [SDOA 99] D. SALBER, A. K. DEY, R. ORR, G. D. ABOWD: Designing for Ubiquitous Computing: A Case Study in Context Sensing. GVU, Technical Report GIT-GVU-99-29, July 1999. <ftp://ftp.gvu.gatech.edu/pub/gvu/tr/1999/99-29.pdf>
- [Schi 95] W. N. SCHILIT: *A System Architecture for Context-Aware Mobile Computing (Ph. D. Dissertation)*. New York: Columbia University, 1995.
- [SATT 99] A. SCHMIDT, K. A. AIDOO, A. TAKALUOMA, U. TUOMELA, K. LAERHOVEN, W. V. VELDE: Advanced Interaction in Context. *HandHeld and Ubiquitous Computing*, Karlsruhe, 1999.
- [SSP 98] T. STARNER, B. SCHIELE, A. PENTLAND: Visual Contextual Awareness in Wearable Computing. *The Second International Symposium on Wearable Computers (ISWC'98)*, Pittsburgh, 1998.
- [UKC] UKC Mobicomp: Mobile and Context-Aware Computing Research at U.K.C. <http://www.cs.ukc.ac.uk/projects/mobicomp/>
- [Wald 99] Waldo, J., "Jini Technology Architectural Overview", White Paper, Sun Microsystems, Inc., January 1999.
- [WDKF 02] ST. WATERHOUSE, D. M. DOOLIN, G. KAN, Y. FAYBISHENKO: Distributed Search in P2P Networks. *IEEE Internet Computing*, Vol. 6, Nr. 1, pp. 68-72, January/February 2002.