A photo management system for future home environments

Qian, Y.; Udink, R.; Feijs, L.M.G.

Published in:
International ITEA Workshop on Virtual Home Environments, February 20 - 21, 2002, Paderborn, Germany

Published: 01/01/2002

C-LAB Publication

Band 12

Ulrich Büker,
Heinz-Josef Eikerling,
Wolfgang Müller (Eds.)

Proceedings
International ITEA Workshop on
Virtual Home Environments

organised by the VHE
Middleware consortium

February 20-21, 2002
Paderborn, Germany

Shaker Verlag
Aachen 2002

A Photo Management System for Future Home Environments

Yuechen Qian[1,2], Rob Udink[2], and Loe M.G. Feijs[1,2]

[1] Eindhoven Embedded Systems Institute,
Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands

[2] Philips Research Laboratories Eindhoven,
Prof. Holstlaan 4, 5656 AA Eindhoven, The Netherlands

y.qian@tue.nl, rob.udink@philips.com, feijs@win.tue.nl

Abstract. In future home environments, users will have access to their digital assets on any device in any room. Home environments will have a high degree of hardware, middleware, software and administrative heterogeneity. Current protocols and systems do not address the consistency problem of mobile data in heterogeneous environments. When data are distributed and duplicated, their logical identities are lost. We propose the introduction of logical identities of mobile data and identity-preserving data exchange. We designed the Universal Resource Access Protocol (URAP) and validated it in the MemorySafe system, an experimental photo management system for future home environments. In this paper, we present the proposed concepts and the URAP protocol. We show how they are used to address the consistency problem of mobile data in the MemorySafe system.

1 Introduction

Nowadays people have all kinds of digital assets at home, such as digital photos, MP3 files and video clips. People can manage their assets on dedicated devices. In the future, home environments will be the places where the inhabitants can access the assets anytime on any device in any room. Not only inside houses, people will also be able to access and share their assets on the move. All digital assets will be available to users on request, as described in Ambient Intelligence [1].

High degree of hardware, middleware, software and administrative heterogeneity is the nature of home environments. The heterogeneity comes from two sources. The first source is the variety of application domains, such as home automation, web-browsing and A/V applications. The second source is the variety of providers of home appliances and applications. The definition and standardization of network service protocols in the application layer of the OSI and TCP/IP reference models [2] is the most suitable foundation to achieve the interoperability among heterogeneous appliances and applications while preserving the openness of the systems for application integration and system evolution.

One crucial element in mobile computing is ubiquitous and consistent data availability. By data we mean both digital assets and files for user profiling, UI customization and user adaptation. Ubiquitous availability of such data can be achieved by the combination of ubiquitous networks and personal mobile computing devices. Seamless integration of both technologies requires data to be transported among storage and computing devices. We called such data *mobile data*. Maintaining the consistency of mobile data in heterogeneous computing environments is crucial to mobile applications, which has not been well addressed in existing data transfer protocols and systems.

When data are distributed and duplicated, their logical identities may get (and often are) lost. Existing data transfer protocols provide locator-based data exchange. Path names in the FTP protocol, URLs in the HTTP protocol, etc. are not reliable mechanisms for client and server sides to determine identities of mobile data. In distributed systems, file identifiers, for example "inodes" in Unix systems, are kept internally unique at the system level. Inter-system data exchange still lacks the notion of data identities.

Therefore, we propose the introduction of logical identities of mobile data and identity-preserving data exchange. We designed the Universal Resource Access Protocol (URAP), which allows identity-preserving data exchange. The URAP protocol is applied in the MemorySafe system, a photo management system for future home environments. The rest of paper is organized as follows. In Sect. 2 we will discuss some related work. We will present the MemorySafe system and the URAP protocol in Sect. 3. We will describe the applied mechanisms to achieve data consistency in Sect. 4. In Sect. 5 we will give some conclusions of our work.

2 Related work

Several distributed file systems have been built to support mobile applications. The Ficus replicated file system [3, 4] is designed for very large scale distributed file systems. Ficus allows updates on replicated data during network partitioning and uses file update propagation and directory reconciliation to resolve conflicts. The Coda system [5] is intended for applications, such as document preparation and program development in mobile environments. Coda improves data availability via caching data and it improves data consistency via using log-based directory resolution [6, 7]. The Roma personal metadata service [8] uses the Universal Resource Names [9–13] schemes to identify mobile data and uses a centralized metadata server to keep track of locations of replicated files.

Some systems adopt middleware-based approaches to mobile computing. The Bayou system [14] supports mobile database-like applications. It allows applications to specify semantic knowledge about conflicts and conflict resolutions. Jini [15] and UPnP [16] are middleware solutions to dynamic environments. They provide mechanisms, such as service discovery, joining, and lookup, to support applications in mobile and dynamic environments. Jini and UPnP do not include specific support to address the consistency problem of mobile data.

Some systems, such as MS Outlook [17] and Lotus notes [18], are developed for dedicated applications. They support mobile users. File synchronizers, such as IntelliSync [19] and QuikSync [20], are user-level programs. Usually those tools do not trace updates on files. In Unison [21], updates are logged.

As can be seen, Ficus, Coda, and Roma are designed for file sharing in professional environments, not in heterogeneous home environments. Using them requires a global Unix file system. Those systems do not address the applications that exhibit high degrees of fine-grain write-sharing, such as photo databases. The Bayou system defines a middleware approach to support database applications in mobile environments. Bayou relies on its underlying platform and does not tackle heterogeneity either. Jini and UPnP do not have built-in mechanisms supporting mobile data. MS Outlook and Lotus notes are designed for enterprise-size environments. Because of their professional use, interoperability and openness are prohibited for security, privacy and commercial reasons. The usability of file synchronizors relies on the complexity of the files in question. They are handy when the number of the files is rather small, the directories in question are rather straightforward, and the number of devices involved is limited. Manually repairing file and directory conflicts is often required.

3 MemorySafe system

The MemorySafe system is a networked photo management system designed for home environments. Fig. 1 illustrates the scenarios of using the system. Users can choose any devices at hand, such as TV screens and web-tablets, to view their photos in the environments. They can also put digital photos into mobile devices, such as PDAs and memory cards, and update those photos on the move. Users can easily synchronize the photos on the mobile devices with those at home.

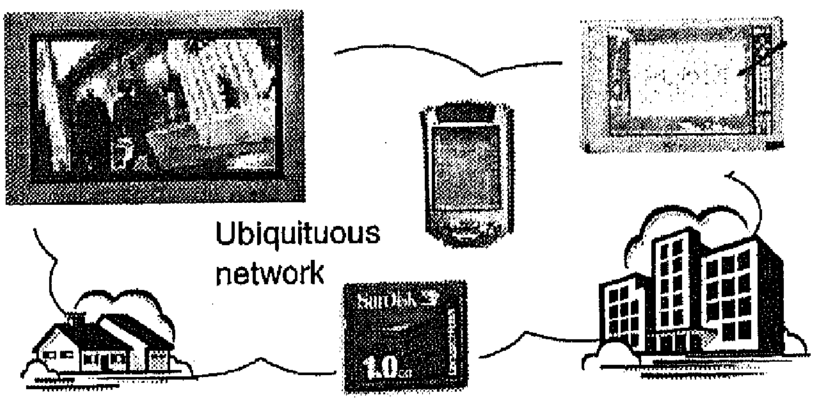

Fig. 1. Scenarios of the MemorySafe system.

The MemorySafe system consists of a number of hand-held or stationary devices, called "MemorySafe hosts". MemorySafe hosts have local storage and wired/wireless connections to local home networks. The communication between MemorySafe hosts is done via the URAP Protocol. Users can browse and manage photos on local and remote MemorySafe hosts.

3.1 Architecture

The MemorySafe system architecture consists of three layers, as illustrated in Fig. 2. In the application layer, applications use resource objects, which represent mobile data entities, such as photos. Resource requests are serviced by resource access objects in the middleware layer. Resource access objects have a common interface, called "resource access interface". The interface can be implemented using different system resources, such as file systems, database systems, and network interfaces in the system layer. In the current system, we have two implementations of the resource access interface, one is based on file systems and the other is based on the URAP protocol.

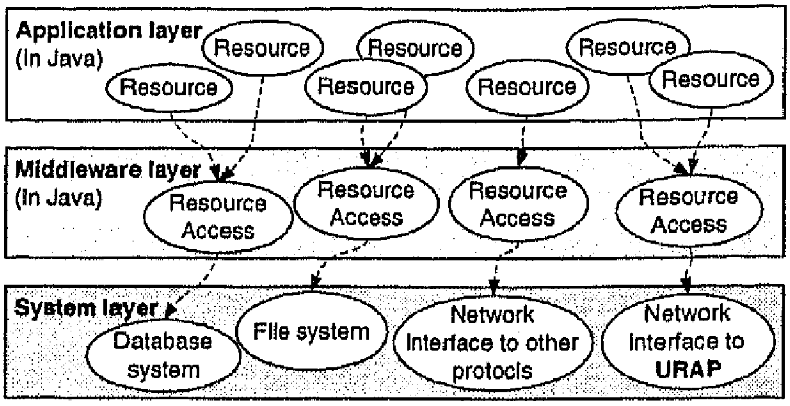

Fig. 2. The system architecture of the MemorySafe system.

3.2 Resources

In the MemorySafe system, mobile data entities are modeled as *resources*, in accordance with the Internet information architecture [9–13]. In the Internet architecture, resources are network accessible information entities existing in devices and on the Internet. Compared with Internet resources, our definition of resources has a broader scope in that data entities in isolated devices and media are also resources as long as they are to be transported among devices. Besides Internet resources, therefore, data entities on storage media such as floppy disks, memory cards, are also resources in our definition. Typical examples of resources include documents, photos, MP3 files, video clips, user profiles, UI setting files.

In mobile and ubiquitous computing, data are often replicated to improve system robustness and performance. Data are cached at client sides to minimize response time and reduce network traffic. Data are stored in portable storage media so that user applications can function when devices are disconnected from the network. Data are archived on permanent storage media for the sake of data safety. From an application's point of view, therefore, data objects may come from different sources, such as HTTP servers, FTP servers, local file systems, storage media, network database systems. There should be mechanisms for reliably identifying and differentiating data entities.

In the Internet information architecture, there is no proper concept to model logical identities of data entities. Internet resources can be identified by universal resource identifiers (URIs). URIs are one-to-one associated with physical resources on the Internet. In other words, URIs are used for describing physical identities of resource. They are not designed to cope with logical identities of replicated resources. Universal resource locators (URLs) are designed for locating physical resources on the Internet. In mobile computing, resources can be distributed into various devices and, therefore, they can be located by different URLs. URLs do not serve the purpose of modeling logical identities of resources.

In mobile computing, modeling logical identities of resources is necessary. From a user's point of view, using logical identities of resources can simplify the view of data in various devices. Imagine a situation where a user puts all his photos in his home PC, part of which on a FTP server and a HTTP site. From the technical point of view, preserving logical identities of resources can better model replicated and duplicated mobile data entities.

In the MemorySafe system, resources are encapsulated in objects with logical identities, descriptions, contents, and histories, as illustrated in Fig. 3. Each resource has a globally unique resource identifier (GURI) as its logical identity, which is persistent, permanent, and immutable. In the implementation, we use the combination of device addresses and timestamps to define resource identifiers. GURIs can be transported with resources into various devices. We call them *identity-preserving operations* the communications in which logical identities of resources are maintained unchanged.

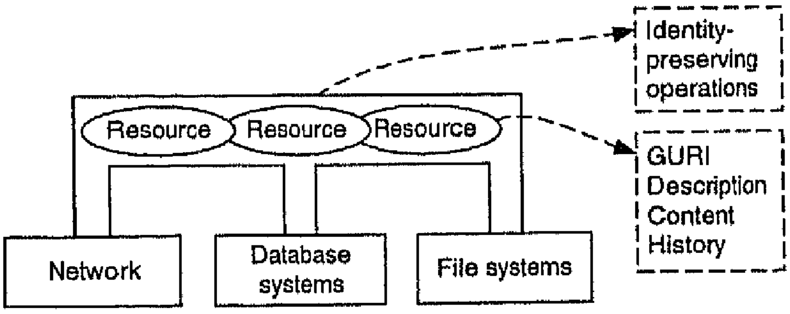

Fig. 3. Resources (mobile information entities).

In the MemorySafe system, resource descriptions are lists of zero or more *resource descriptors*. A resource descriptor is a pair of a *key* and a *value*. Resource descriptions record metadata of resources. Resources have histories. Because of partial and disconnected updates on mobile data, updates must be logged for later data synchronization.

3.3 Universal Resource Access Protocol

To support resource-based communication and identity-preserving operations, we propose the Universal Resource Access Protocol (URAP). Fig. 4 indicates its positions in the OSI and Internet reference models.

The URAP protocol is very similar to the FTP protocol, as they share the same model. The main difference between the two protocols is the objects being transferred. In the FTP protocol, the processed objects are files, which in

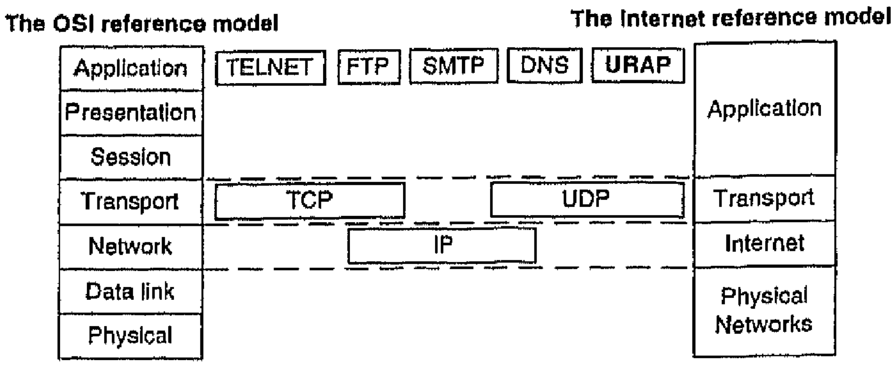

Fig. 4. The URAP protocol in OSI and Internet reference models.

our point of view are lacking logical identities. In the URAP protocol, the processed objects are resources, which are self-contained objects including logical identifiers, rich descriptions, and access histories. Moreover, the URAP protocol includes primitives to query resources.

Fig. 5 illustrates how resources are handled in the URAP protocol. Applications can use pathname-like names to retrieve resources. Applications can also search for resources on specified criteria. Applications can selectively access resources, resource descriptions, resource contents, resource histories. The URAP protocol allows identity-preserving data exchange over the network. To achieve the consistency of replicated and duplicated resources, transaction-like mechanisms can be integrated in applications.

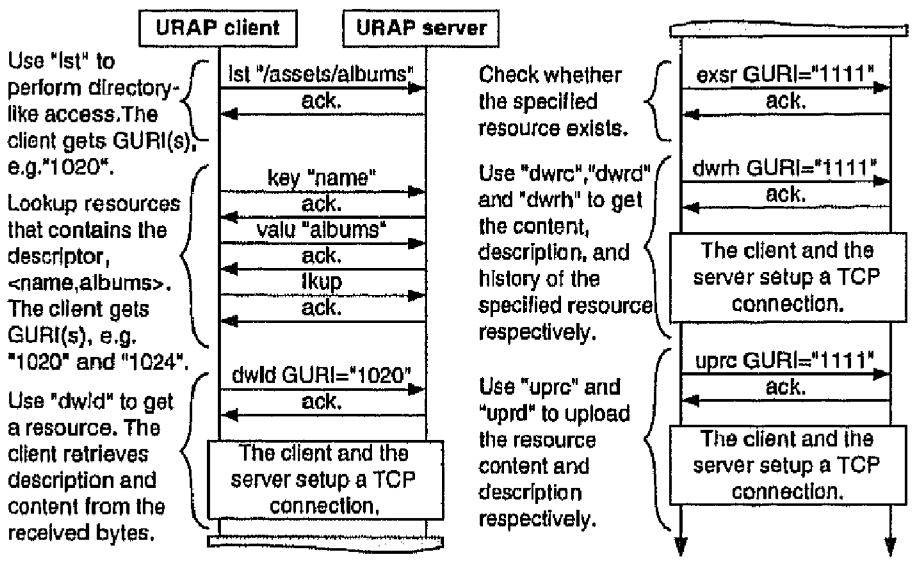

Fig. 5. The resource-based communication in the URAP protocol.

4 Resource consistency

In the MemorySafe system, users can use a GUI tool, called "PhotoExplorer", to manipulate their photos. For example, users can move photos from one device to another on the interface by dragging photo icons and dropping them onto the icon representing the target device.

In the MemorySafe system, two semantics of the drag-n-drop operation are defined, as illustrated in Fig. 6. One is so-called "identity-preserving", which means the created photo resource in the target device has the same identity as the operated photo resource in the source. The other is so-called "creating", which means the photo resource in the target device has a newly-assigned GURI. Thus photo resources on both sides have different identities. By default the identity-preserving semantics is applied.

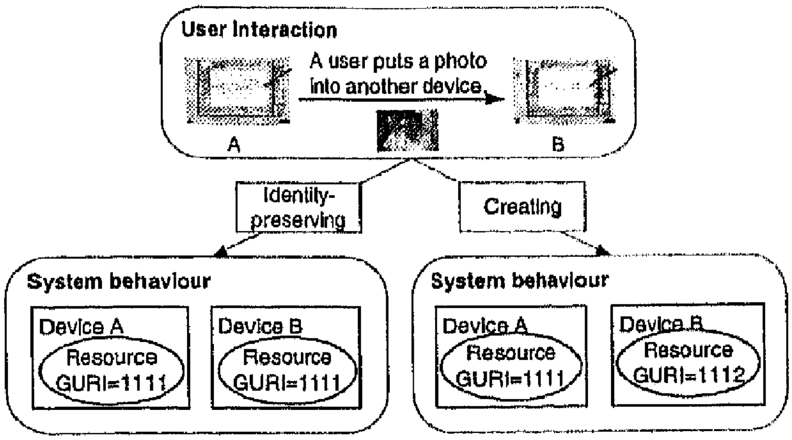

Fig. 6. Two semantics of the drag-n-drop operation on the user interface.

When users perform identity-preserving drag-n-drop operations, it can often happen that the photos being moved exist already in target devices. In those circumstances, resource synchronization is triggered to resolve inconsistencies.

For each resource in the MemorySafe system, modifications on its content and description are recorded in its history. Each time the content is modified, a version number is assigned and logged in the history. Insertions and removals of descriptors and modifications on their values are logged as well. Each entry in the resource history has five fields, `timestamp`, `operation`, `descriptor`, `pre-value`, and `post-value`. The fields of `descriptor`, `pre-value`, and `post-value` are defined by users and applications. The `timestamp` field is filled in by the system. The `operation` field depends on the action performed. Its value is either "setDescriptor" or "removeDescriptor".

In the file-based synchronization [4, 6], the first type of conflicts to be resolved is name-related, such as:

- Create/Create conflicts. A pathname is used for creating different files.
- Rename/Rename conflicts. A file and its copy are renamed.
- Update/Rename conflicts. A file is updated while its copy is renamed.

In the MemorySafe system, the introduction of GURIs helps to resolve those conflicts. The logical name of a resource is regarded as a presentation property and only modeled as a part of the resource description. Resources are allowed to have the same names. Thus, create/create conflicts are completely avoided in the system. Rename/update conflicts are resolved by simply propagating each operation on the other side. Rename/rename conflicts are resolved by applying the latest rename operation on both sides.

In the file-based synchronization, update/update conflicts need to be resolved as well. For example, a file and its copy are updated in their devices. In the MemorySafe system, update/update conflicts of resources having the same GURI have two forms.

- Set/Set conflicts. A descriptor is assigned with different values in devices.
- Set/Remove conflicts. A descriptor is assigned with a new value in one device while the same descriptor is removed in another device.

Updates on resource contents are expressed by updating *lastest_version*, a built-in descriptor of resources. By serializing resource histories, set/set conflicts can be resolved by applying the latest "set" operation on both sides. In handling a set/remove conflict, the "removal" operation is ignored and the "set" is applied on both sides, for the sake of data integrity.

Another type of conflicts are directory inconsistencies. For example, two directories in question do not have the same content. In the MemorySafe system, *collections* are the resources that contains other resources. A collection contains the GURIs of its element resources, i.e. its content is a list of GURIs. Collections are analogous to directories in file systems, but differ from directories in that they only contain "links" to their element resources. Demolishing a collection does not damage its element resources. In the MemorySafe system, photo albums are modeled as collections. Collections are firstly treated as normal resources in the synchronization. In case of content updates, special treatments are taken. Content updates of collections are insertions and removals of GURIs of element resources. To resolve content inconsistencies between two collections, the GURIs of the element resources of both collections are merged and propagated to both sides for the sake of data safety and integrity.

5 Conclusion

In this paper, we presented our proposal of introducing logical identities and identity-preserving operations to cope with the consistency issues of replicated mobile data in mobile computing environments. Due to the high degree of heterogeneity of home environments, the definition and standardization of service protocols helps to achieve the interoperability and preserve the openness. We presented the URAP protocol, which realizes identity-based access and identity-preserving operations of mobile data. We presented the MemorySafe system for managing photos in home environments. The MemorySafe system uses the URAP protocol to achieve the consistency of mobile data.

Acknowledgements

The MemorySafe system has been designed and implemented within the "Perceptive Home Environments" project [22], which was jointly launched by Eindhoven Embedded Systems Institute and Philips Research Laboratories Eindhoven (Nat.Lab.). The authors acknowledge the members of the project: E. Dijk, E. van den Hoven, N. de Jong, E. van Loenen, D. Tedd and D. Teixeira. We would like to acknowledge the ITEA Ambience project [23] for its support and E. van Loenen and M. Bodlaender for their comments on the paper.

References

1. Denning, P.J.: The Invisible Future: The Seamless Integration of Technology into Everyday Life. McGram-Hill Professional Publishing. (2001) 235–257
2. Tanenbaum, A.S.: Computer Networks (3rd Edition). Prentice Hall, Inc. 1996
3. Guy, G.R., Heidemann, J.S., Mak, W., Page Jr., T.W., Popek, G.J., Rothmeier, D.: Implementation of the Ficus Replicated File System. Proceedings of the Summer USENIX Conference 1990. (1990) 63–71
4. Reiher, P., Heidemann, J., Ratner, D., Skinner, G., Popek, G.: Resolving File Conflicts in the Ficus File System. Proceedings of the Summer USENIX Conference 1994. (1994) 183–195
5. Braam, P.J.: The Coda Distributed File System. Linux Journal(50). (1998)
6. Kumar, P., Satyanarayanan, M.: Log-Based Directory Resolution in the Coda File System. Proceedings of the Second International Conference on Parallel and Distributed Information Systems 1993. (1993)
7. Kumar, P., Satyanarayanan, M.: Flexible and Safe Resolution of File Conflicts. Proceedings of the USENIX Winter 1995 Technical Conference. (1995)
8. Swierk, E., Kiciman, E., Laviano, V., Baker, M.: The Roma Personal Matadata Service. Proceedings of the 3rd IEEE Workshop on Mobile Computing Systems and Applications. (2000)
9. Moats, R.: URN Syntax. Network Working Group RFC 2141. (1997)
10. Sollins, K., Masinter, L.: Functional Requirements for Uniform Resource Names. Network Working Group RFC 1737. (1994)
11. Berners-Lee, T.: Universal Resource Identifiers in WWW: A Unifying Syntax for the Expression of Names and Addresses of Objects on the Network as used in the World-Wide Web. Network Working Group RFC 1630. (1994)
12. Berners-Lee, T., Fielding, R., Masinter, L.: Uniform Resource Identifiers (URI): Generic Syntax. Network Working Group RFC 2396. (1998)
13. Berners-Lee, T.: Masinter, L., McCahill, M.: Uniform Resource Locators (URL). Network Working Group RFC 1738. (1994)
14. Terry, D.B., Theimer, M.M., Petersen, K., Demers, A.J., Spreitzer, M.J., Hauser, C.H.: Managing Update Conflicts in Bayou, a Weakly Connected Replicated Storage System. In SIGOPS 1995. (1995)
15. Sun Microsystems, Inc.: The Jini Technology. http://www.sun.com/jini/
16. Microsoft Corp.: The Universal Plug and Play. http://www.upnp.org/
17. Microsoft Corp.: Microsoft Outlook. http://www.microsoft.com/office/outlook/
18. IBM Corp.: Lotus Notes. http://www.lotus.com/homepage.nsf
19. Pumatech, Inc.: Intellisync. http://www.pumatech.com/intellisync.html
20. Iomega, Inc.: QuickSync. http://www.iomega.com/intellisync.html
21. Pierce, B.C.: Unison file synchronizer. http://www.cis.upenn.edu/ bcpierce/unison/
22. Koninklijke Philips Electronics N.V.: Perceptive home environments. http://www.research.philips.com/generalinfo/special/phenom/
23. Information Technology for European Advancement(ITEA): The ITEA AMBIENCE project. http://www.extra.research.philips.com/euprojects/ambience/