

A Middleware Framework for the Adaptive Home

H. P. Dommel¹, R. Wagner, R. Edwards, and A. Doran
Department of Computer Engineering, Santa Clara University, USA

Abstract. We discuss the concept of adaptive middleware for augmented living spaces, with the aim to provide a framework to systematically outfit homes with the flexible intelligence to cooperatively optimize their shared resources. To exemplify this concept, we discuss the EMISS testbed and platform for rapid design, simulation and deployment of wireless sensor networks for the “adaptive home” beyond off-the-shelf control mechanisms for networked home appliances. Based on a combination of web and object technology, the system offers location-independent, user-friendly access to customized energy-aware home control, as well as the basic ambient intelligence to make a home adapt to user behavior and environmental constraints.

Keywords. Network middleware for smart homes, sensor information tracking, contextual data filtering and fusion, adaptive resource management.

1. Introduction

Ambient intelligence for smart homes aims to improve the comfort or sustainability of living spaces through background technology. Various frameworks on communication mechanisms have been proposed in the past for distributed, real-time and embedded applications to provide seamless, resource-aware middleware services [1], for example for home multimedia systems using home network middleware architectures such as Jini, VHN, or Universal-Plug-and-Play (UPnP) [2].

However, despite a plethora of academic and industrial approaches to enrich homes with such capabilities, an adaptive middleware framework for characterizing the design and deployment of sensor network support for assistive living spaces is still lacking. Available solutions widely differ in implementation and capabilities. This paper fills the gap by discussing a basic methodology and platform to deploy network middleware to retrofit living spaces with adaptive intelligence and dynamic service quality provisioning, for example to regulate energy consumption based on user patterns and environmental measurements. The rest of the paper is structured as follows: Section 2 discusses related work. Section 3 outlines foundations for such adaptive middleware and discusses design criteria for a testbed for rapid prototyping. Section 4 characterizes the EMISS architecture as an instance of a platform based on this framework, and Section 5 offers an outlook.

¹ Correspondence to: H. P. Dommel, Department of Computer Engineering, 500 El Camino Real, Santa Clara University, Santa Clara, CA 95053-0566, USA; Tel.: +1 408 554-4485; Fax: +1 408 554-5474; E-mail: hpdommel@scu.edu

2. Related Work

Intelligent systems in a home environment receive information such as continuous feeds from a number of dispersed devices, typically sensors arranged in a sensor network. The information streams must be analyzed, communicated both internally and externally, and take configuration input from users to regulate device behavior. Newer developments in outfitting homes with powerful telematic intelligence [3, 4] catering to their dwellers' critical living needs go beyond the basic premises of networked home appliances, and introduce challenges such as context-awareness [5], Quality-of-Service support, and multi-device adaptation. Making homes intelligent, aware servants to their inhabitants has been a common theme in previous work on the home of the future [6, 7]. In contrast to control bus systems such as Luxmate or EIB, which use centralized control and lack intelligence, smart approaches include for example neural network support to anticipate user needs and conserve energy [8]. In the wake of a new generation of ambient intelligence applications and services [9], programming and configuring home-embedded systems with respect to the physical properties is a major challenge. Major questions concern the placement of adaptive capabilities in intelligent environments to recognize users and their operational needs, predicting adjustments in real-time to these needs based on contextual knowledge, and allowing access to control services and features in an unobtrusive, yet tangible and secure manner. We argue that middleware, as the coordinative and reusable software "glue" residing between applications, the network protocol stack and underlying operating systems, should provide this adaptivity. Other middleware architectures for smart homes have been proposed, with a different focus on platform independence [10], or universal and seamless interoperability [11]. Other recent work tackles the problem specifically from a context-awareness perspective [12].

3. Adaptive Home Middleware

Adaptation is a central ability of any natural organism or artifact to react to changes in the environment. Middleware, as reusable software residing between applications, operating systems, network protocols and devices, bridges the gap between lower-level hardware and software infrastructure. For smart homes, next-generation middleware shields developers from tedious low-level details, supports development of applications with consistent capabilities, offers high-level interface abstractions by detaching applications from sensors, and makes integration with heterogeneous software components simpler. We observe that to date only ad-hoc, application-specific solutions have been proposed for smart home middleware to provide some degree of adaptation. Over time, home telematics will become increasingly fine-grained and sophisticated, in terms of stringent requirements for dependability, trustworthiness, predictability, reconfigurability, awareness, accuracy, scalability, and efficiency. It is hence necessary that a service-quality-aware middleware supports multiple dimensions of service provisioning, encoded in meta-data as a function of device and sensor capabilities, and enforces them to meet the various end-to-end requirements between devices in the "adaptive home". If implemented properly, such services will elevate application development from low-level platform details such as sensor specifics, providing a higher-level abstraction for distributed resources and a consistent set of capabilities close to given design-level specifications.

The design rationale behind this architecture follows the premise of simplicity, with benefits for end users, future system upgrade, and uncomplicated deployment. For ease of use, the system should operate with legacy wireless technologies. For ease of integration, the system should provide a turn-key solution to automatically configure a smart home environment. For easy of maintenance and low cost, the system should use off-the-shelf components for microcontrollers and sensors. The system should be extensible to new measurement and control components to optimize collective resource consumption in living spaces.

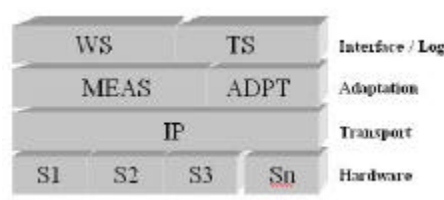


Figure 1. Adaptive Home Middleware Layers.

Fig. 1 depicts the basic layered structure of such adaptive middleware, with an array of sensors S_i at the bottom layer communicating via IP with the above measurement and adaptation modules, which again feed into a transaction server TS for data conversion, logging and queries. A web server WS provides role-based, authenticated access to control data. Middleware refers here to network mediated measurement and adaptive control, independent of a specific implementation toolkit such as Jini.

4. EMISS Architecture

The EMISS (Energy Monitoring via the Internet and Sensors for Sustainable living) project illustrates the idea to provide a modular and simple platform for the design and deployment of a wireless sensor network for the adaptive monitoring of information appliances. The resulting system serves as a testbed for rapid prototyping and testing of adaptive home environments, based on the notion of a component framework [13], and a reconfigurable middleware platform.

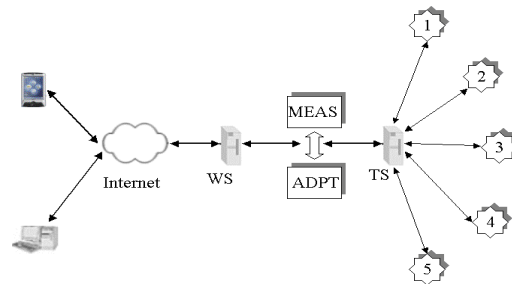


Figure 2. Principal EMISS architecture.

The principal architecture of EMISS consists of several basic components: an array of sensors, a transaction server (TS), and a web server (WS), as illustrated in Fig. 2. TS acts as sensor manager and aggregating relay between sensor-enabled devices, receiving wireless data transfers of active sensors and integrating updates in its measurements (MEAS) database. It contextualizes measurements, derives recommendations for new settings, and after feedback from the user, sends adjustments through a transformation engine back to the devices. Based on user-specified constraints and sampled usage patterns, current settings are recalculated in the adaptation component (ADPT) to reflect a desired or preferred energy consumption pattern and usage profile for the house. A knowledge base hosted in TS serves as repository for raw measurements and meta data from past and current sensor input. At the frontend, EMISS provides a sensor dashboard and sockets-based interface for manual or automatic device control. Users can access sensor information and control devices remotely via a personalized web interface provided by WS. This scaling web-interface allows users to monitor their smart home remotely from any web-enabled device, hiding the complexity and heterogeneity of the underlying technicalities.

4.1. Components

Basic hardware requirements in EMISS include a set of sensors, a sensor bus, an intelligent monitoring system and control units, with the premise that the system can be extended modularly. For the sake of transparent connectivity among devices and ease of deployment, EMISS uses wireless communication to interconnect devices. At the core of EMISS is a control unit (“EMISS device”) with a microcontroller and a communications controller, connected to a power source and a transceiver linking wirelessly to appliances. This unit connects to MEAS and ADPT modules and interfaces with the transaction server TS and the web server WS. In a simpler design, all these functions converge in the EMISS device.

Sensors provide the data necessary to determine power intake, allowing for example to maintain a power consumption profile for connected devices. The gathered sensor inputs on variables such as light, temperature, or noise are periodically communicated through a preprocessor to TS, which uses PHP scripts to write the data to a MySQL database. In this process it performs key conversions from raw sensor data to strings of name-value pairs for storage in the database. TS relays the data via TCP to the web server WS through the MEAS and ADPT modules. These updates are converted by the transaction server to nine different name-value pairs and are appended into an URL for access through WS, which represents new device states as flags in web pages. This information is then parsed by TS and sent back to the core device for regulating the individual device states. The website allows users to interact with specific devices, for example to control the power being used by the appliances. Although our implementation uses TCP, EMISS does not prescribe a specific underlying transport protocol to interface sensors with TS. Device polling in the current implementation is in the order of minute intervals. If home profiles are to be accessed from external sites, security in terms of authentication and encryption will be a paramount concern. The software core of EMISS is depicted in Fig. 3, depicting the flow of communication within the embedded software and front end components.

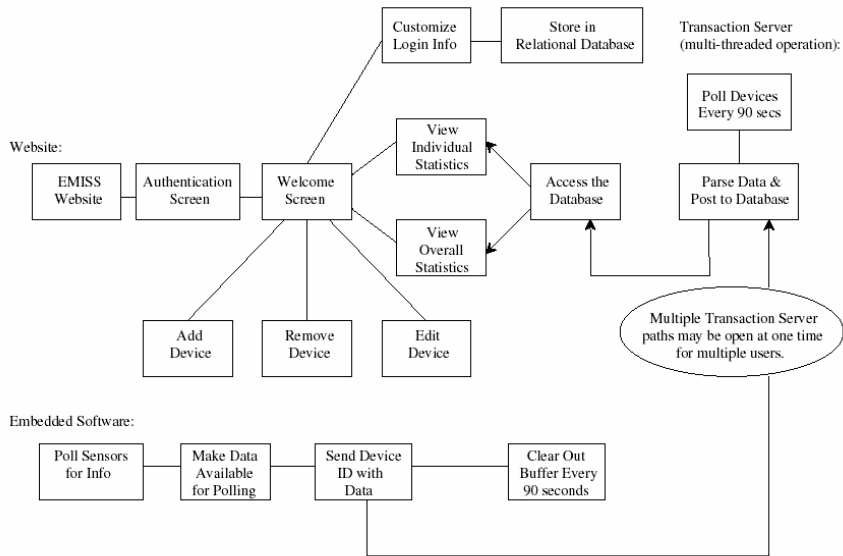


Figure 3. Software components in EMISS.

EMISS is implemented with Java (except backend communication with sensors), with benefits for portability and usage on a broad spectrum of devices. The generated traffic is lightweight, which makes the system suitable even for low-end wireless access. Per measurement exchange, packets from the core device contain at maximum 60 bytes of data.

Traditionally, it is difficult to configure systems that simultaneously can satisfy multiple service quality properties. For greater flexibility, EMISS supports the notion of a component-based, hierarchical, adaptive, and reflective middleware architecture, which allows to characterize smart home platforms depending on application domains, multi-modal device characteristics and customized capabilities, such as specific security features. Sensory applications in EMISS are focused on sustainable, environmental measurements, but extensions to physiological or lifestyle monitoring are feasible. Although currently not implemented, distributing the adaptation intelligence across several subsystems would increase greater confidence in data and sensor validation. For greater scalability, EMISS components could be associated into local grids to orchestrate a larger array of sensors, or larger grids to cooperate on wide-area energy monitoring.

4.2. Adaptation

It is a common view that the major obstacles in the introduction of smart home technologies are users' problems in handling complex devices, and resistance to technological innovation. Design of simple user interfaces may solve one part of the equation, but equally important is the incorporation of adaptive capabilities into smart home environments and their user interfaces to improve usability and lessen the need

for users to configure and control their smart home. User adoption of this technology is hence a matter of building adaptation to the user into the environment. User-centered adaptivity and its proper placement within service layers is challenging due to the many functions and control procedures inherent to smart environments. In our design, adaptation is perceived as a middleware feature placed between the user interface and the underlying device and network services, with benefits in modularity and flexibility. The design of adaptive mechanisms may take into account various methodologies, including user modeling in artificial intelligence, control theory, learning algorithms from neural networks, pattern recognition, or prediction algorithms [14]. A simple algorithm for adaptation to usage patterns considers previous measurements in a weighted relation to new measurements:

$$NewValue = (1 - W) * PreviousValue + W * SampledValue$$

where $0 < W < 1$ (usually very small) is an inertia weight for currently sampled values in relation to previous settings to prevent major oscillations in the adaptation process. Remotely adjustable devices are regulated accordingly based on the new values. Adaptation is not a monolithic process that applies uniformly to all components. The equation has to be fine-tuned to the metrics and measurements incurred with the various sensors, given environmental conditions, time and place. Adjustment of devices is a cooperative process, in order to optimize the collective settings of devices and their feedback effects on the adaptive home surroundings.

5. Outlook

We have discussed an application and component-based middleware framework for systematically deploying adaptive capabilities into the smart home of the future. The underlying motivation is to bypass rediscovery and redevelopment of software components for such environments. As a bare-bones case study, the EMISS project has been presented as a toolkit and platform to turn a regular home into an energy-aware smart home. The EMISS system exists currently as a prototype in our research lab. We plan to integrate it into a new building on our campus showcasing sustainable design. Future work entails refinement of system design, precision of measurements and feedback to end users, interoperability among hardware components, maintenance, and security. More sophisticated data mining of measurements in relation to user preferences may yield more precise resource control and adaptation in energy-aware homes. Other future features include ambient intelligence for user presence and voice recognition [15], context awareness, and greater adaptation to the user in intuitive and transparent ways. Aside from pure engineering aspects, socio-technical aspects such as dependability and usability will be critical if end-users are to adopt such environments for every day use [16]. In conclusion, we perceive our system as a useful building block in shaping telematic services for smart homes into “killer applications” of future sustainable living spaces.

References

- [1] M. Roman and R. H. Campbell. A Middleware-Based Application Framework for Active Space Applications, *Lecture Notes in Computer Science*, Volume 2672, pp. 433 - 454, Jan. 2003.
- [2] R. Gupta, S. Talwar and D. P. Agrawal. Jini Home Networking: A Step toward Pervasive Computing, *IEEE Computer*, V. 35, N. 8, pp. 34-40, 2002.
- [3] B. Brumitt, B. Meyers, J. Krumm, A. Kern and S. Shafer. EasyLiving: Technologies for Intelligent Environments, In *Proceedings of the Second International Symposium on Handheld and Ubiquitous Computing (HUC 2000)*, pp. 12-29, Sept. 2000.
- [4] M. Coen. Design Principles for Intelligent Environments, In *Proceedings of the 15th Nat'l Conf. Artificial Intelligence*, AAAI Press, Menlo Park, Calif., pp. 547-554, 1998.
- [5] A. K. Dey, G. D. Abowd and D. Salber. A Context-Based Infrastructure for Smart Environments, *Managing Interactions in Smart Environments*, Springer-Verlag, pp. 201-213, 1999.
- [6] C. Douligeris. Intelligent Home Systems, *IEEE Communications Magazine*, Vol. 31, No. 10, pp. 52-61, Oct. 1993.
- [7] S. S. Intille. Designing a Home of the Future, *IEEE Pervasive Computing*, Vol. 1, No. 2, pp. 76-82, Apr/Jun 2002.
- [8] M. C. Mozer. An Intelligent Environment Must Be Adaptive, *IEEE Intelligent Systems*, Vol. 14, No. 2, pp. 11-13, Mar/Apr 1999.
- [9] T. Basten, L. Benini, A. Chandrakasan, M. Lindwer, J. Liu, R. Min, and F. Zhao. Scaling into Ambient Intelligence, *The 6th Design, Automation, and Test in Europe Conference (DATE'03)*, Munich, Germany, pp. 76-83, March 2003.
- [10] T.-Y. Ku, D.-H. Park, and K.-D. Moon. A Java-Based Home Network Middleware Architecture Supporting IEEE 1394 and TCP/IP, *IEEE Transactions on Consumer Electronics*, Vol. 48, No. 3, Aug. 2002.
- [11] K.-D. Moon, Y.-H. Lee, Y.-S. Son, and C.-K. Kim. Universal Home Network Middleware Guaranteeing Seamless Interoperability among the Heterogeneous Home Network Middleware, *IEEE Transactions on Consumer Electronics*, Vol. 49, No. 3, Aug. 2003.
- [12] M. C. Huebscher, J.A. McCann. Adaptive Middleware for Context-Aware Applications In Smart-Homes, In *Proceedings 2nd Int. Workshop on Middleware for Pervasive and Ad-Hoc Computing*, Canada, October, 2004.
- [13] C. R. G. de Farias, M. van Sinderen, and L. F. Pires. A systematic approach for component-based software development. In *Proceedings of the Seventh European Concurrent Engineering Conference (ECEC'2000)*, Leicester (UK), pp. 127-131, April 2000.
- [14] S. K. Das, D. J. Cook, A. Bhattacharya, E. O. Heierman Iii, and T.-Y. Lin. The Role of Prediction Algorithms in the Mavhome Smart Home Architecture, *IEEE Wireless Communications*, p. 77 - 84, Dec. 2002.
- [15] A. Gárate, I. Lucas, N. Herrasti, and A. López. Ambient Intelligent Technologies for Home Automation and Entertainment, In *Proceedings of the Second European Symposium on Ambient Intelligence (EUSAI)*, Eindhoven, NL, 2004.
- [16] R. Schaefer and H.-J. Eikerling. Increasing the Acceptance of Ambient Intelligent Technologies for Wellbeing at Home through Security Contexts, In *Proceedings of the Second European Symposium on Ambient Intelligence (EUSAI)*, Eindhoven, NL, 2004.