Final paper: 6th World Multi-Conference on Systemics, Cybernetics and Informatics, **SCI 2002**, July 14-18, 2002, Orlando, Florida, USA

Internet Embedded Control Networks – A Reality Check

Tom Pfeifer
FOKUS (Fraunhofer* Institute for Open Communication Systems)
Kaiserin-Augusta-Allee 31, D-10589 Berlin, Germany
pfeifer@fokus.fhg.de

Hermann Hartenthaler
T-Systems Nova GmbH / Berkom
Goslarer Ufer 35, D-10589 Berlin, Germany
hermann.hartenthaler@t-systems.de

* GMD FOKUS joined the Fraunhofer-Gesellschaft, FhG, in July 2001

ABSTRACT

A production-level implementation of an integrating architecture for accessing various infranets via intranets and the Internet as well as telecommunication networks is based on an analysis of the heterogeneous systems for interconnecting distributed infrastructural devices, such as low-bandwidth sensor/actuator-networks. It follows the experience from a research prototype. The modularity of the architecture allows the rapid deployment of new application scenarios. However, the scalability is critically discussed, leading to an outlook of further IP-fication of the control devices.

Keywords: Embedded Systems, Ubiquitous Computing, Web-based computing, Office Control, Java-based network programming, Infranet, Intranet, Applications of Distributed Systems, Network Reliability, Network Security

1. Introduction

Integration of building-controlling infrastructure networks ('infranets') into wider intranets and the Internet has often been discussed recently within the context of bringing Mark Weiser's [1] idea of Ubiquitous Computing into reality.

Benefits are expected for remote Facility Management, e.g. for organizations running office buildings and/or collecting resource consumption parameters from residential facilities; as well as for users, i.e. people working in offices and living in homes being able to check and control functions remotely, automate daily routines, and employ comfortable multimedia edge-devices for home surveillance.

Many promising demonstrators have been seen in the labs [4][7][9], very few descriptions of practical realizations could be found [10][11].

The authors had the opportunity to transfer the prototypical approaches from the research environment (at Fraunhofer FOKUS) into industrial practice of the representational headquarters of Deutsche Telekom AG. The implementations have been done by FOKUS spin-off company Ivistar AG.

The system discussed within this paper is on production level, i.e. the third generation. It follows the first, research level prototype within FOKUS [6], and the second, a small-scale trial system within the Deutsche Telekom.

The Building

The Representational Headquarters of Deutsche Telekom was built in 1999-2002, referred within this paper as "the building". It provides a forum for the development of innovative visions for the future, for discussions about the trends in culture, politics, science, technology and society. The communication and presentation technologies provided within the building contribute to a multi-medial experience of a new kind.

Interactive, intelligent networking is a major focus of the underlying infrastructure within the building. Innovative guidance and information systems individually lead the visitor through the whole house. Technologies representing the state of the art in industry and research are integrated and combined, forming a complex, but functional, flexible and easy-to-use structure.

The Task

FOKUS provides technological consulting and contributes to the planning process in areas of multimedia and telecommunication requiring innovation, combining and integrating state-of-the-art technologies to obtain new effects. Implementations by Ivistar bring leading edge communication technology from research to reality.

The task was to make the infranet sub-systems within the building reachable from a variety of end-systems for different purposes, focusing on user and operator-specific control of the building. Infranet technology used within the building, besides IP based systems, mostly consists of LON and EIB based control networks (cf. section 2), proprietary sub-systems for specific tasks, and a LON based active badge system [12][13] for location dependent services.

Why are these systems so heterogeneous, even in a new building? In our research projects in 1999 we claimed that it would be impossible to find construction companies in the different trades who could use a unified sub-network technology, currently. Despite FOKUS's influence on the architectural planning process towards cutting edge technology, our claims proved to be true. Each branch of trade depends on the support of component manufacturers and their own experience. Further, they guarantee the functionality of their products only for a closed set of properties. The only way to handle this complexity is to introduce a new integration layer on top of all these sub-systems.

This sub-system integration, which is the topic of this paper, has to interwork (on operational level as well as database sharing) with other innovative systems, such as smart IP devices and location-dependent information and communication systems.

The integration has to hide the complexity from the users, which are departments and companies renting office space, and which are organizers of events such as conferences, large meetings, concerts, public relations happenings. Abstracting from underlying technology and heterogeneity, these users should be presented an object oriented view, bundling controllable elements in their natural relation. Preferred adjustments, here called scenes, need to be stored and re-called.

In contrast to the closed concept of most sub-systems, this approach has to provide an open concept and unified interfaces

to users and applications. Our goal was to establish a modular platform for Deutsche Telekom, focusing on the integration of infranet technology, already installed or newly planned, into Internet and intranet scenarios.

This paper discusses a brief selection of related work in section 2, followed by the discussion of requirements and the description of the implemented architecture in section 3, user interfaces in section 4, followed by future application examples and critical conclusions in section 5.

2. Technological Background

An estimated number of "traditional" communication end-systems, nearly 500 million computers and 800 million telephones, are connected either to the Internet (understood as the full-size-computer interconnection with worldwide access), intranets (the same type of interconnection with more restricted access), or phone networks.

Beside, there are already more than 20 *billion* sub-computer devices equipped with micro controllers [3][7]. Such devices are used in nearly every area of actual life, ranging from car engine control, heating systems, video cassette recorders, alarm and surveillance systems, elevator control, room access restriction, light scene settings, household appliances, up to the whole area of industrial automation.

These devices collect and process an enormous amount of information. However, as they are either not connected to networks at all or only to networks dedicated to the specific application environment (e.g. an automation process), they are not able to share the gathered information or their processing results. Such sub-computer network controlling infrastructure is called an *infranet* here.

Each sub-computer system works nicely on its own, however, their interaction would enable an impressing number of new applications. The current relevance is driven by the existence of thousands of such infranets already installed, connecting millions of legacy devices.

Automation and Building Networks

Automation networks are quite heterogeneous, often very proprietary, partly traditional, partly innovative. They range from serial lines and buses (RS232, RS422), industrial control networks (Fieldbus, AnyBus), building automation networks (LON [14], EIB [15], InstaBus, HomeRun [17], CEBus [18], X-10), to newer sub-computer wired and wireless links and networks (USB, FireWire [23][24], WLAN [26], Bluetooth [25], IrDA [20]). Few approaches try to bridge or harmonize such networks, often only within the same category.

Networking technology is shortly introduced for cases which are not as widely known as IP supporting networks, and the current wireless approaches mentioned above.

The Local Operation Network (LON) developed by EcheLON [14], supports free topology twisted pair cabling for 78 kbit/s, backbone structures for 1.25 Mbit/s or power line networks. Introducing the Neuron chip with three pipelined micro controllers, it runs a proprietary communication protocol and application software. The "LonWorks Network Services" provide a multi-client / multi-server-platform for installation and maintenance.

The European Installation Bus (EIB) [15] is a system for home and building automation in free topology with a bitrate of 2.4 kbit/s. EIB is supported by more than 100 companies in 15 countries. InstaBus is the brand name from Siemens, BatiBus is a similar French version. The European Home System (EHS), developed by European manufacturers, provides a low-cost plug-and-play network (various low voltage cables, power line, IR, radio) with an open, layered technology and object-oriented command language. There are alliances towards EIB and other systems.

The Controller Area Network (CAN) has been developed mainly for usage in cars, but is used today for many automation purposes [16]. Its CSMA/CA (Collision Arbitration) serial bus with multi-master and real-time capabilities allows inherent prioritizing (e.g. motor control commands have priority to convenience). A payload of 0...8 bytes can be transmitted with up to 1 Mbit/s in networks up to 40 meters. Decreasing the bit rate allows 1000 meters with 50 kbit/s.

Power line transmission approaches are part of most of these systems described above, enabling access to devices in already existing buildings that cannot be reached with low-voltage bus lines. This advantage of using the mains distribution wires is paid for with higher cost of the controllers and additional measures in the electrical installation for line couplers and surge protection and facing severe EMC problems and concurrent usage models.

Integration Approaches

Some selected integration approaches that have influenced our work, are referenced briefly.

The Universal Plug and Play Forum (UPnP) [22], an industry group of companies led by Microsoft, promotes networking protocols and device interoperability standards for home and small business.

The Java Intelligent Network Infrastructure (Jini) [21], initiated by Sun Microsystems to provide a generic application environment for easy interworking of devices has found its way into the Java enterprise environment.

Home Plug and Play (HomePnP) [19] is an extension of the CEBus [18] and its CAL Standard (Common Application Language, EIA-600 and EIA-721) for interoperability of sub-systems.

The EURESCOM project P915/HINE – Heterogeneous Inhouse Networking Environment [9] has developed a platform for communication, home automation, control and facility management applications and services; and implemented an Internet integrated test-bed for (pre-)commercial components, applications and services.

Brumitt [4] describes "technologies for intelligent environments" and underlines the important role of a middleware, connecting networked stand-alone devices, so that these devices could continue working even if central server component fails.

3. System Architecture

The infranet control system ("VistaControl"), as described in this paper, is part of a product portfolio implemented on a common middleware platform by Ivistar. Other applications on this platform focus on location dependent visitor information and guidance ("VistaRoom"), room booking combined with active door-plates ("VistaDoor"), etc., sharing appropriate resources.

The necessity of object-oriented middleware platforms, defining sets of principles and components supporting openness, flexibility, and programmability, has gained general acceptance within the recent years. It provides an abstraction from the complexity of the underlying structure of heterogeneous hardware platforms, operating systems, and the difficult networking functionality. CORBA is our current choice for vendor openness.

For the operating system, GNU/Linux was chosen for all servers. Beyond the well-known advantages of an Open Source operating system, we had the best insight into the requirements of the low-level drivers for connecting the infranet sub-systems.

Java was chosen as platform and implementation language for all parts of the system (except the low-level drivers) for portability, object orientation and good network programming support.

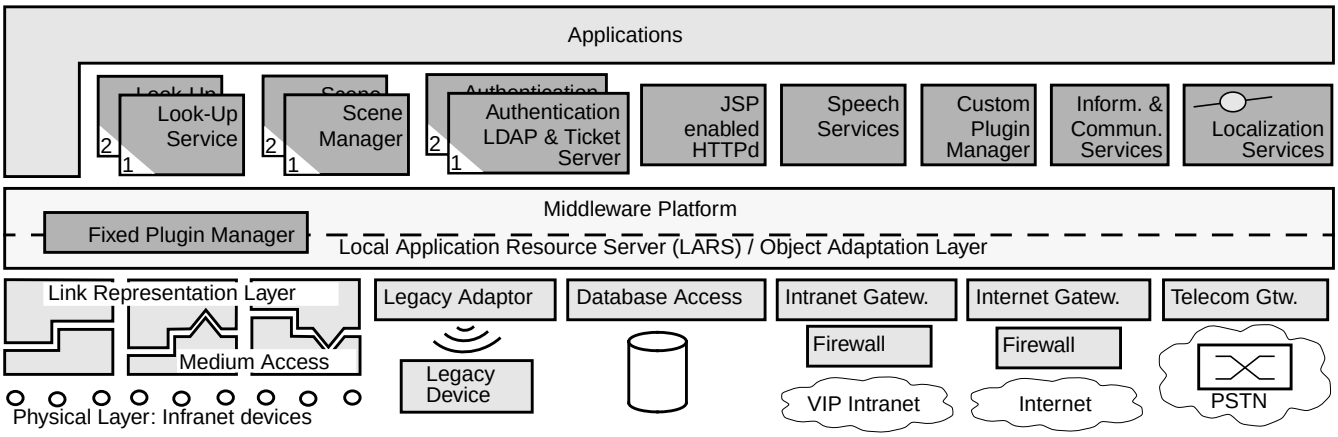

Figure 1 Software Architecture Overview for the Internet – Intranet – Infranet Integration Platform

Prior to the description of the system modules, the approaches for robustness, authentication, and scalability are discussed now.

Robustness

A production level system requires specific considerations. As soon as we build not a toy or a lab prototype, and the work processes of people depend on it, the robustness of a system is a key factor for usability as well as acceptance by the users. In the case discussed within this paper, the system is not only used for more convenience in office environments, but also for the smooth and reliable course of events with hundreds of invited guests.

Providing reliability is always a trade-off between optimizing cost, in order to keep the system affordable, and adding system components for redundancy, supervision and management.

While preparation activities, such as editing scenes, can rely on repair contracts within 24 hours; proper reliability within the course of events can only be provided by cold or hot redundancy, thereby avoiding a “single point of failure” for critical processes.

This section only discusses the reliability of the system built on top of the existing, reliable infrastructure (such as the underlying Ethernets, backup power and redundant power supplies).

Core servers of the system are duplicated (primary and secondary server) in order to provide hot redundancy. While several software servers run on the same physical computer, they are duplicated and distributed the same way, so that failure of one piece of hardware leaves the environment for executing scenarios in a working state.

The interface computers have multiple connections to the segments of the infranets. This is necessary for performance reasons, on the other hand the native (slow) links within the infranet segments take over some traffic if one of the interface cards fails.

Authentication

An important safety feature within a large building is the authentication of users who want to control something. This authentication process should not be too complicated in order to keep the whole system acceptable, in particular for non-technical people. Everybody would get annoyed by entering a password before being able to dim the light in the own office, and use the manual light switch instead. Further, the system maintenance should be affordable. [5]

Therefore, several levels of security are provided, depending on the task to perform. Controlling the light, shading and temperature within an office has less consequences and requires less security than doing the same thing for a VIP meeting room or even for the event hall. In our approach, we distinguish *Host Based Authentication* and *User Based Authentication*. Both methods are supported by a *Ticket Manager*, as follows.

Host Based Authentication: Host based authentication employs the hostname and/or IP address to recognize specific workstations. This approach serves for lower level security when a fixed relation exists between a stationary workstation and a room to be controlled, which is the typical situation for desktop workstations in offices. It is independent of the user logged into this workstation, providing an advantage for shared workstations and guests within a room. Access rights based on host names are very easy to configure in servers. The major advantage is that this authentication process is the least annoying for the user.

This approach assumes, that a user who is able to access a workstation would also be able to use the manual controls within the same room (light switches, heating valves), in particular with single-user operating systems. For multi-user systems allowing remote login, the application could be restricted to the user who owns the console, or rely (for this low security level) on the social behaviour of the user (while violations could be logged).

The host based authentication in general is susceptible to IP spoofing or misconfiguration. In a firewalled intranet, this is not a problem regarding outside attacks. Inhouse IP faults can be answered by the network administrator properly.

Recognition of the MAC address of the Ethernet cards increases security, thereby allowing also users with portable computers to participate in host based authentication. However, access rights based on MAC addresses are more difficult to configure in the server software, and a place easily been forgotten to modify when equipment is replaced. A much better, central place to map MAC to IP addresses are the switches for the local Ethernet. As they are usually in a physically locked room together with the patch panel of the structured cabling system, this approach provides a very reliable mapping of specific hosts to specific rooms.

User Based Authentication: User based authentication serves for all cases where host based authentication is either to insecure or otherwise not appropriate, in particular for remote control functions. This approach has to consider that in larger companies, computational rights of any kind are typically managed within centralized directories, accessed via the LDAP protocol [27] in the specific case. Many directory servers on the market are LDAP enabled, making it an ideal choice for vendor neutral directory access.

The authentication process begins with requesting name and password from the user, or can be combined with the normal daily workstation login. It can be supported by other means of person recognition, such as chip cards, active badges [12] [13], transponders, or biometric procedures. These supportive methods can, on the one hand, provide additional security, on the other hand determine the location of the user within the building.

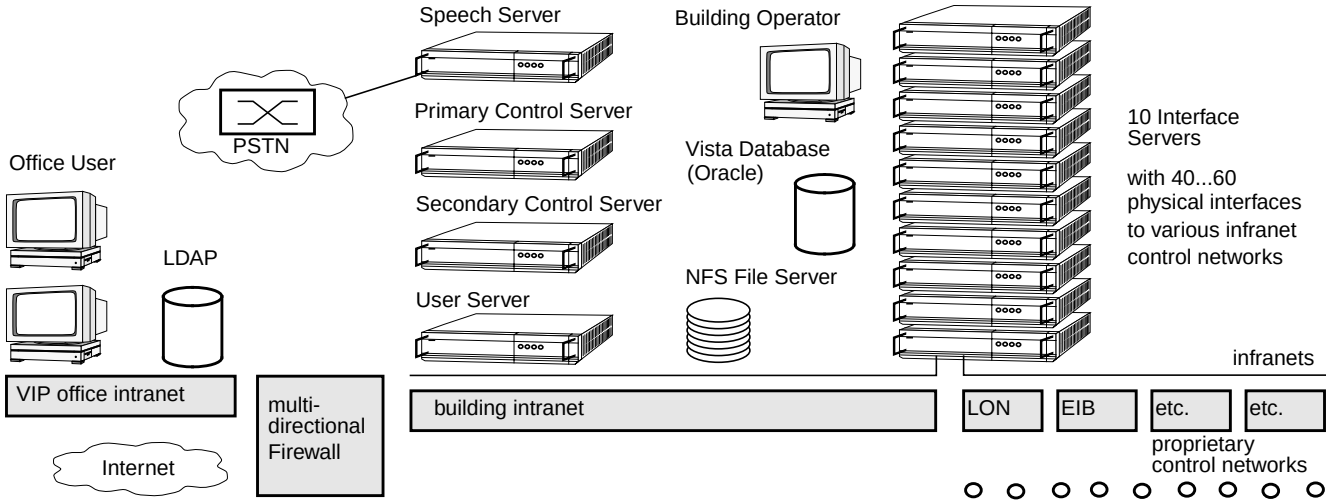

Figure 2 Hardware Architecture Overview

Authentication Tickets: A specific problem to discuss is that the authenticated user (either host based or user based) triggers building control processes which run longer and independent of the login session. Further, these processes should have permissions differently grained than the user structure within the company. E.g., a technical operator might own not only the pass key to all door locks in the building, but also the right to control any building function. When he has just triggered the predefined run of scenes for the 4-hours event in the main hall, and leaves now to set some functions in the meeting room, he wants to avoid an – even accidental – influence to the running event. User owned processes, as typical in Unix systems, are still too coarsely grained and thereby not sufficient for this task.

An additional aspect is that in large scenarios, there is a tremendous number of control interactions, which would overload the LDAP server shared with other services.

A solution is provided with a ticket based system. The *Ticket Manager* is the central component for management and validation of access rights within the building control system. Client applications, asking for access to the system, have to register proper account data with the Ticket Manager. The latter reads the access rights via LDAP, and issues a ticket for the client, which it has to present for any further access to the system. Server applications can check the access rights against the Ticket Manager using this ticket, thereby restricting the set of possible interactions. The same user can be issued different tickets when he uses different client applications and when he selects different tasks within the same application.

Scalability

The system is scalable from a variety of viewpoints. The connection of further infranet segments requires additional cards in the interface servers, and additional servers if all card slots are occupied, thereby also distributing the infranet traffic. Further types of infranets can be easily integrated, and legacy devices can be supported. Some restraints are discussed in section 5.

The performance of the control servers is sufficient for large office buildings. The system is also down-scalable to one single computer for small, non-critical environments.

From the application point of view, the platform allows the easy creation of further value added services by combining collectable information and control functions, some examples will be given in the outlook.

Software Architecture

The requirements discussed above lead to an architecture as depicted in Figure 1. The interconnecting element among all components is a CORBA based object oriented middleware, providing an abstraction of the physical distribution and network configuration.

Essential components are duplicated as primary and secondary servers for redundancy reasons. When one server is dysfunctional, other parallel running servers can take over the functions. Main components of the system can be identified as follows.

The **Look-Up Service** provides the users of the system (Intranet Applications and Java Server Pages) with references to the required objects, which represent the functions of the real infrastructure components. Search templates with the required properties are passed to the Look-Up Service, which returns a list of currently available services. Such properties are, e.g., “all devices in room 5002”, “all lamps”, “all dimmable lamps”, or “all LDAP servers”.

Initially, the **Authentication, LDAP and Ticket Server** authenticates the user against an LDAP server. If the authentication succeeds, the Ticket Server fetches the building-control access rights of the authenticated user via LDAP and issues a ticket which validates the session of the user. Using this ticket, the Ticket Server is able to decide whether the owner has the right to employ a specific resource at a specific point of time, for each inquiry of the Look-Up Server or the Scene Manager.

The **Scene Manager** manages, maintains and stores scenarios, which are maintained in object oriented descriptions. Using the Scene Manager, device profiles such as room temperature and brightness, are editable and can be saved as personal adjustments in profiles. This allows the user to control whole groups of devices with a single, predefined action. Additionally, time-dependent animations can be defined this way, as far as the actuators (lamps, displays) allow.

There are two ways to create the scenes. Within the intranet, or from trusted remote networks, browser-based or stand-alone applications can be used to directly influence the equipment and test the scenes. Without connection to the real system, a graphical editor can define scenes based on downloaded room profiles, and upload the edited scenes later to the infrastructure database.

Two **Plug-In Managers** are deployed in different layers, one for *Fixed Plug-Ins*, and one for *Custom Plug-Ins*. The plug-ins are required to create virtual objects, i.e. representations of virtual sub-systems.

Previously known routines within the same sub-net are implemented as *fixed plug-ins* in order to keep processes on the lowest possible level for the most performant execution. For example, to trigger the light in a room by the nearby motion detector would be implemented as fixed plug-in, which is able to appoint the task to the infranet devices directly. Such executions will not generate any traffic in upper service layers.

Custom Plug-Ins are used in any cases which are not predefined, and in any cases where rules span several sub-networks or involve external resources.

Plug-Ins are also used to define closed control circuits, e.g. temperature guided heating control, which are independent from access right restrictions.

The **Infrastructure Network Communication Layer** provides interfaces and gateways towards all the different physical infrastructure networks and autonomous proprietary systems.

The degree of proprietarity and heterogeneity of infranet systems leads to a small protocol stack, where the Medium Access is specific to the network. Often, these networks have their own logical tier, mapping network devices to logical items like virtual shared memory or virtual network variables. These logical tiers are harmonized in the Link Representation Layer, providing object oriented interfaces towards the middleware platform.

Legacy adaptors provide access to devices which are not networked at all, but provide a remote control interface, e.g. consumer electronics with IR control.

Because in the discussed building LON is the infranet used for all light and shading control systems, thereby the largest of the infranets there, this example is used for a few technical details:

LON cards (physical layer), plugged into the interface servers, are supported by a Linux device driver, delivering LON messages in raw form (medium access). The link representation layer processes these messages, considers the LON-specific “Standardized Network Variable Type” (SNVT) and presents the extracted, typed information in object oriented form. Device-dependent properties are hidden under a unified interface.

The **Local Application Resource Server** (LARS) component provides a container to collect various objects for communication layer interfaces and for the fixed plug-ins. This component is contained in every software package where necessary, thereby avoiding single points of failure.

The resources bundled in LARS interwork with the underlying communication modules via internal CORBA interfaces. In their combination they represent a distributed communication layer, providing an aggregation of information from the individual modules, and presenting “abstract appliances”. On top of LARS, there are only manufacturer and network independent models of appliances, represented by the sum of their individual properties. These properties can be queried and set by applications, restricted by the respective access rights.

The **Database Access** component provides an object oriented view on underlying relational databases. They are accessed via JDBC, are shared with other services (employee, room and event information; room booking; communication services), thus allowing seamless integration of building control functions with building wide information and communication systems.

The **Java Server Pages** (JSP) enabled **HTTP Daemon** provides the interface for simple control functions, and for external users (outside the building intranet) restricted by the firewall to use of port 80 communication. JSP technology provides a convenient method to integrate dynamic content into HTML pages. The widely used, reliable Apache HTTP server is combined with the TomCat JSP/Servlet engine [31].

Information, Communication and Localization Services are used for the location based communication, information and guidance system, which is beyond the scope of this paper.

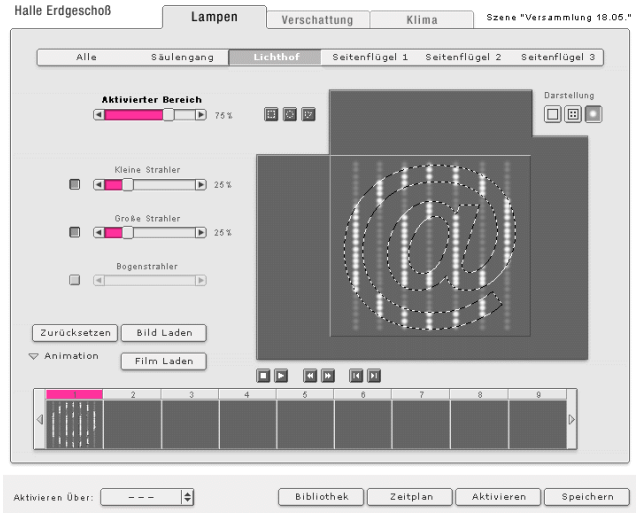

Figure 3 Graphical control applets (e.g. lights)

All **Applications** can use the HTTP based access as described above. Within the building intranet, application programs can access the middleware platform directly. Third party applications can employ specifically implemented custom plug-ins.

A **Management** console continuously displays the status of the complete system and informs about occurring technical problems. The status of the individual components is provided via the Internet standard protocol SNMP [28][29][30], making this information also accessible by a centralized network management solution of the building.

Hardware Architecture

Depicted in Figure 2, the *Primary and Secondary Control Server*, respectively, host the duplicated software components, namely the Look-Up Service, the Primary Scene Manager, the Primary Authentication, LDAP and Ticket Server. The *User Server* hosts the JSP enabled HTTP daemon, the Custom Plug-In Manager, the Middleware component for custom database connectivity, and the Interface packages for proprietary sub-systems. The *Interface Servers* host the Local Application Resource Server (LARS), the LON and EIB communication modules, as well as the Fixed Plug-In manager with the Plug-Ins.

4. User Interfaces

This section can only provide a brief overview of the variety of possibilities provided to the users for controlling their environment, focusing on the implementation for Deutsche Telekom.

Speech and WAP based interfaces have been implemented for demonstrational use, only. The reliability of a speech activated control highly depends on the recognition engine.

The **GUI-based control applications** are divided into such limited to firewall-permitted HTTP port 80 access (dynamically generated HTML-pages only), for office control; and rich and complex applications with full access to the middleware and the supporting components, for controlling complex technical systems such as event halls.

With a *Lightweight Office Control*, running in any web browser, office users can control lights, shading, heating, and air conditioning in their own rooms as well as meeting rooms. They can predefine rules and scenes, which can later be recalled manually or triggered by timers or external events.

A main goal of the *Rich Control Applications* for complex, heterogeneous technical systems is the usability for non-technical people, maybe in the creative trade, and occasional personnel.

Therefore, the GUI and the operating sequences should be as intuitive and simple as possible.

Platform independent Java applications have been developed to create scenarios on-line as well as off-line. This means, that *device configurations profiles* for a specific room in XML can be downloaded and sent to the design office. The creative people develop lighting scenarios and send the respective XML *scenario files* back.

The graphical editor allows different ways of definitions. Single lamps or groups of them can be touched with the mouse to edit the percentage of brightness, or a bitmap image can be overlaid the geometrical structure of lamps, in order to display graphics (e.g. an "@" in Figure 3). Similarly, air conditioning and shading can be edited.

Back in the building, the XML files can be uploaded, and with a test application the scenarios can be tested in single step or group modes, thereby allowing fine-tuning of the scenes.

In order to be able to re-call the complex scenes with a large number of infrastructure actuators, they cannot be activated sequentially (approx. 100 ms per actuator) in a performant way. Therefore, they must be downloaded to the actuators into their scene registers, and re-called with broadcast commands, limiting the high-level traffic. These activations can be issued from a computer applet as well as from a wall switch or small tableau.

The advantage the whole system provides, is that now the limited number of actuator registers can be refilled for each event with individual scenarios, which otherwise would have been defined once for the lifetime of the building.

5. Summary and Evaluation

Faced with already existing installations of heterogeneous infranet technology, we built a production-level platform for the integration of such infranets and gateways for infranet, Internet, and telephony remote access.

The system enables the creation of new service modules, leading to rapid deployment of new services for application scenarios/environments.

Scenarios for usage-dependent control and billing for facilities can be built easily. As people produce heat, the number of active badge wearers plus passively detected room users control heating/air conditioning. The used meeting room is automatically scheduled for next night cleaning, avoiding cost for cleaning unused rooms. The life cycle of the projection light bulb is monitored and can be changed before failure. The room usage of the ad-hoc meeting is billed to the project of the responsible person.

Evaluating the results of the installation, however, we found the cost paid for integrating any kind of sub-network was higher than expected, despite positive effects of modularity.

Real-time behaviour becomes less and less predictable with increasing complexity of inter-dependent layers.

The hardware effort on the interface side would not be affordable for a normal office building. The interface servers practically form an additional backbone-skeleton to the underlying infranets, which scales at least linear with the latter, and beyond linearity when the number of simultaneous actions of actuators exceeds their limited scene storage capacity.

In summary, there is a mismatch between cost and added value. To solve this problem, we currently focus on leaner solutions, which involve "Smart IP devices", i.e. sub-computer nodes with fully integrated IP stack. These nodes can be embedded in "Zero Gateway" architectures.

The next office buildings to be built by Deutsche Telekom will provide a large installation base, allowing to produce smart IP nodes in high quantities, leading to manufacturing prices below the current LON or EIB nodes.

The advantage of our platform is that this migration can be done transparently for applications and users, and proprietary legacy devices can still be supported within the transition period.

6. References

[1] Weiser, Mark: Some Computer Science Issues in Ubiquitous Computing. - Communications of the ACM, 36(1993)7, July 1993, pp. 75-84

[2] Estrin, Deborah; Govindan, Ramesh; Heidemann, John: Embedding the Internet. - Communications of the ACM, 43(2000)5, May 2000, pp. 38-41

[3] Tennenhouse, David: Proactive Computing. - in: Communications of the ACM, 43(2000)5, May 2000, pp. 43-50

[4] Brumitt, Barry; et al.: EasyLiving: Technology for Intelligent Environments. - in: Thomas, Peter; Gellersen, Hans W. (Eds.): Proc. of Handheld and Ubiquitous Computing, Bristol, UK, Sep 25-27, 2000. - Berlin, Heidelberg, New York: Springer, 2000

[5] Link, Carsten; Luttenberger, Norbert: Sicheres Nomadic Computing in Intranet-Umgebungen – Problemstellungen und Lösungskonzepte. - Proc. Kommunikation in Verteilten Systemen, KiVS, Hamburg, Germany, Feb. 20-23, 2001. - Berlin: Springer, 2001

[6] Pfeifer, Tom: Internet - Intranet - Infranet: A Modular Integrating Architecture. - Proc. 7th IEEE Workshop on Future Trends of Distributed Computing Systems, FTDCS'99, Cape Town, South Africa, Dec. 20-22, 1999; Los Alamitos: IEEE Comp. Soc. Press

[7] Luckenbach, Th.: Seamless Integration of Infranetworks into the Internet: The I-Cube-C Project. - in: Proc. of the Home Networking 11053, London, 14-15 Sep. 1999

[8] Popescu-Zeletin, R.; Pfeifer, T.: A Modular Location-Aware Service and Application Platform. - Proc. 4th IEEE Symp. on Computers and Communications, ISCC'99, Red Sea, Egypt, July 6-8, 1999

[9] Project P915-PF. HINE – Heterogeneous In-house Networking Environment. Deliverable 4. Description and evaluation of the HINE demonstrator. - Heidelberg: EURESCOM, June 2000

[10] Smarthome Forum: http://www.smarthomeforum.com/

[11] Gesytec - Expansion of LON networks: http://www.gesytec.com/

[12] Harter, A.; Hopper, A.: A Distributed Location System for the Active Office. - IEEE Network, 8(1994)1, Jan/Feb. 1994, IEEE Computer Society, pp. 62-70

[13] EIRIS Infrared Localization System. System Manual. - ELPAS Electro-optic Systems Ltd., Raanana, Israel, 1999

[14] LonWorks Engineering Bulletin. - Echelon: Palo Alto, 1995

[15] European Installation Bus (EIB); http://www.eiba.com

[16] Controller Area Networks; http://www.can-cia.de/

[17] Home Phoneline Networking Alliance. http://www.homepna.org/

[18] CEBus Standard EIA-600; http://www.cebus.org/cebus.htm

[19] Home Plug & Play; http://www.cebus.org/hpnp.htm

[20] Infrared Data Association (IrDA); http://www.irda.org/

[21] Jini Technology. Sun Microsystems: http://java.sun.com/jini

[22] Universal Plug and Play Forum (UPnP); http://www.upnp.org

[23] IEEE Std 1394-1995. IEEE Standard for a High Performance Serial Bus. - Piscataway, NJ, 1995.

[24] IEEE Draft 1394b - Long Distance Serial Bus, 1999

[25] Bluetooth Wireless Technology: http://www.bluetooth.com

[26] IEEE Standard 802.11. Wireless Local Area Networks (WLANs), IEEE group P802.11; http://grouper.ieee.org/groups/802/11/

[27] Lightweight Directory Access Protocol (LDAP), http://www.ietf.org/rfc/rfc1777.txt

[28] Structure and Identification of Management Information for TCP/IP-based Internets, http://www.ietf.org/rfc/rfc1155.txt

[29] Management Information Base for Network Mgmt. of TCP/IP-based internets: MIB-II http://www.ietf.org/rfc/rfc1213.txt

[30] Simple Network Management Protocol (SNMP) http://www.ietf.org/rfc/rfc1157.txt

[31] TomCat JSP Engine: http://jakarta.apache.org/tomcat/