Internet-integrated Building Control: Leaving the Lab – Robust, Scalable and Secure

Tom Pfeifer
Fraunhofer[1] Institute for Open Communication Systems (FOKUS)
Kaiserin-Augusta-Allee 31, D-10589 Berlin, Germany
pfeifer@fokus.fhg.de

Andreas Micklei
Ivistar Kommunikationssysteme AG
Ehrenbergstraße 19, D-10245 Berlin, Germany
andreas.micklei@ivistar.de

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

Abstract

Based on an analysis of the heterogeneous systems for interconnecting distributed infrastructural devices, such as low-bandwidth sensor/actuator-networks and the research prototype experience, the paper describes a production-level implementation of an integrating architecture for accessing various infranets via intranets and the Internet as well as telecommunication networks. Its modularity allows the rapid deployment of new application scenarios.
Keywords: Web-based computing, Ubiquitous Computing, Office Control, Java-based network programming, Infranet, Intranet, Applications of Distributed Systems, Network Reliability, Network Security

1. Introduction

The rapid progress in the convergence of computing and telecommunication technologies is set to globally transform the fundamentals of commerce, politics, and culture – redefining the way we all live, work, learn, and play. New application scenarios are enabled within shorter and shorter periods of time. Mark Weiser's idea of Ubiquitous Computing [1] breaks into reality within this decade.

Beside "traditional" communication end-systems, such as maybe 500 million computers connected to the Internet, and nearly 800 million telephones, there are already more than 20 *billion* sub-computer devices equipped with micro controllers [3][31]. These 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 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 [4], EIB [5], InstaBus), to newer sub-computer wired and wire-less links and networks (USB, FireWire [13][14], WLAN [16], Bluetooth [15], IrDA [10]). Few approaches try to bridge or harmonize such networks, often only within the same category.

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 networks already installed, connecting millions of legacy devices. Finally, we cannot wait until every refrigerator has its IPv6 number.

In{ter|tra|fra}net, the tongue twisters distinguished by one or two letters respectively, should here be understood as follows (cf. Figure 1, based on [31]):

- the Internet – the full-size-computer interconnection with worldwide access,
- an intranet, the same type of interconnection, but with restricted access,
- an infranet – a sub-computer network controlling infrastructure devices.

Within this context, Ivistar Kommunikationssysteme AG and FhG FOKUS have established a modular platform for Deutsche Telekom, focusing on the integration of infranet technology, already installed or newly planned, into Internet and intranet scenarios.

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

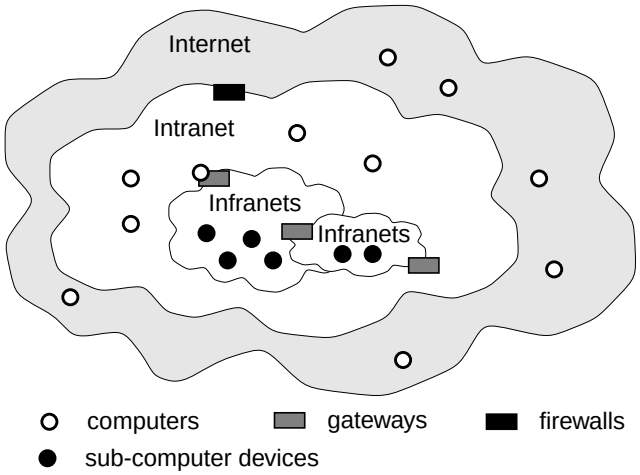

Figure 1 Internet – Intranet – Infranet hierarchy

The project is based on sound experience within the FOKUS institute and the commercial spin-off company Ivistar regarding system integration, location aware technology, middleware platforms, Personal Communication Support (PCS), filtering and dynamic conversion of communication media, and mobile agents.

This paper discusses the application environment next, a brief selection of related work in section 3, followed by the discussion of requirements and the description of the implemented architecture in section 4, user interfaces in section 5 and future application examples in section 6.

2. Production Level System for Deutsche Telekom

The system described within this paper is on production level, i.e. the third generation. It follows the first, research level prototype within FOKUS [26], and the second, a small-scale trial system within the Deutsche Telekom.

Their Representational Headquarters is being built in 1999-2001 (1st unit finished I/2000), referred within this paper as “the building”. It is planned to be a forum for the development of innovative visions for the future, for discussions about the trends in culture, politics, science, technology and society.

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 and easy-to-use structure, allowing the flexible usage of all facilities. The communication and presentation technologies provided within the building contribute to a multimedial experience of a new kind.

FOKUS provides technological consulting and contributes to the planning process in areas of multimedia and telecommunication that require innovation and combine and integrate state-of-the-art technologies to obtain new effects; implementation by Ivistar brings 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, proprietary sub-systems for specific tasks, and a LON based active badge system [28][29] 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. FOKUS supported the architectural planning process of this building by technology consultancy. Despite this influence 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 (e.g. active door-plates) and the “Vista-Site” [34] 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.

3. Related Work

3.1. Automation and Building Networks

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

The Local Operation Network (LON) developed by EcheLON [4], supports free topology twisted pair cabling

for 78 kbit/s, backbone structures for 1.25 Mbit/s or power line networks. It introduced the Neuron chip with three pipelined micro controllers, running a proprietary communication protocol and some application software. The "LonWorks Network Services" provide a multi-client / multi-server-platform for installation and maintenance.

The European Installation Bus (EIB) [5] 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 [6]. Its CSMA/CA (Collision Arbitration) serial bus with multi-master and real-time capabilities allows inherent prioritizing (e.g. functional commands for brakes or motor control have priority to convenience, like window operation). 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.

Further systems for building and industrial automation are CEBus [8], X-10, HomeRun (Home Phoneline Networking Alliance) [7], Fieldbus, AnyBus, etc.; as well as traditional serial lines and buses (RS232, RS422).

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. While it is planned to support up to several Mbit/s (facing severe EMC problems), the available products support relatively low bit rates only. Additionally, these approaches have to be coordinated individually with similar activities, such as power plant control signals or Internet access to the curb.

3.2. Integration Approaches

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

The Java Intelligent Network Infrastructure (Jini) [11], initiated by Sun Microsystems to provide a generic application environment for easy interworking of devices has found its way into the Java enterprise environment.

The Universal Plug and Play Forum (UPnP) [12], an industry group of companies led by Microsoft, promotes networking protocols and device interoperability standards for home and small business.

Home Plug and Play (HomePnP) [9] is an extension of the CEBus [8] and its CAL Standard (Common Application Language, EIA-600 and EIA-721) for interoperability of sub-systems.

The EURESCOM project P915 – Heterogeneous Inhouse Networking Environment (HINE) [30] provided an overview of key technologies for the realisation of heterogeneous in-house networking systems for the home and office environment. It developed a platform for communication, home automation, control and facility management applications and services; and implemented an Internet integrated test-bed for commercial and pre-commercial components, applications and services.

Brumitt [23] describes "technologies for intelligent environments" and underlines the important role of a middleware, connecting networked standalone devices, so that these devices could continue working even if central server component fails.

4. System Architecture

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. Applications may directly access this middleware, or employ services for commonly required functionality. CORBA is our current choice for vendor openness.

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 ("VistaSite"), room booking combined with active door-plates ("VistaRoom"), etc., sharing appropriate resources.

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.

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

4.1. 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 increased convenience in office environments, but for reliable function 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 existing infrastructure, which is considered as constructed reliable enough for the tasks in the building and is out of scope of this paper. Therefore, topics as the reliability of the underlying Ethernets, the Uninterruptable Power Supply (UPS) and redundant power adapters within servers, the backup mains power of the building are not discussed here.

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 can take over some traffic if one of the interface cards fails.

4.2. 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 if one had to enter a password before being able to dim the light in the own office, and use the manual control (e.g. the light switch) instead. Further, the system configuration should not be too difficult to maintain. [25]

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 described as follows.

4.2.1. 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 as least annoying as possible for the user in everyday situations.

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 (where violations could easily 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, while inhouse IP faults could be answered by the network administrator properly [21].

Recognition of the MAC address of the Ethernet cards increases security. 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, thereby allowing also users with portable computers to participate in host based authentication.

4.2.2. 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. While the kind of this directory is of no particular importance for the system described, it is accessed via the LDAP protocol [17] in the specific case. LDAP is formalized in various

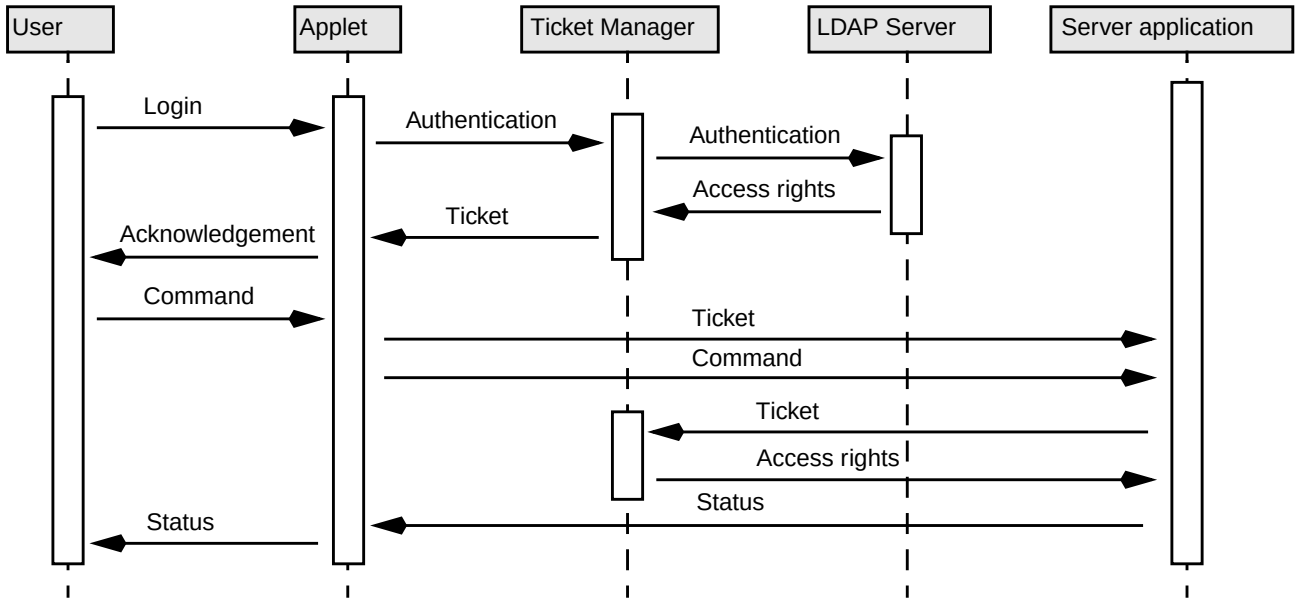

Figure 2 Ticket supported authentication

draft and proposed internet standards. Many directory servers on the market are already LDAP enabled, making LDAP an ideal choice for vendor neutral directory access. However, the Ticket Manager could be easily adapted to access a legacy directory service which is not yet LDAP enabled, if desired.

The authentication process begins with requesting name and password from the user. It 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 [28] [29], transponders, or biometric procedures. These supportive methods can, on the one hand, provide additional security, on the other hand provide the location of the user within the building.

4.2.3. 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. For example, 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, such 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, as presented in Figure 2. 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 with the Ticket Manager, providing proper account data. The latter reads the access rights via LDAP from the central directory service, 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.

Example 1: Paul Freshman calls the office control applet on his PC, without a specific login. The applet calls the Ticket Manager and sends the IP address. The Ticket Manager maps the IP address to the respective room-based account, asks via LDAP for the rights of this account, creates and returns a ticket to the applet. The applet shows the ticket at all control requests to the server applications. Because the ticket determines the rights granted to the user, the applet itself can restrict its graphical user interface to these devices in the room the user is allowed to control.

Example 2: Lara Responsible connects a laptop to a socket somewhere in the building. She calls the control applet for lighting and shading of the main hall, and enters her login name and password. The applet sends these data to the Ticket Manager, which reads the rights from LDAP and issues the respective ticket. Lara adjusts her settings. Later, she switches to another plane of the applet for preparing some scenes in the meeting room. While her applet

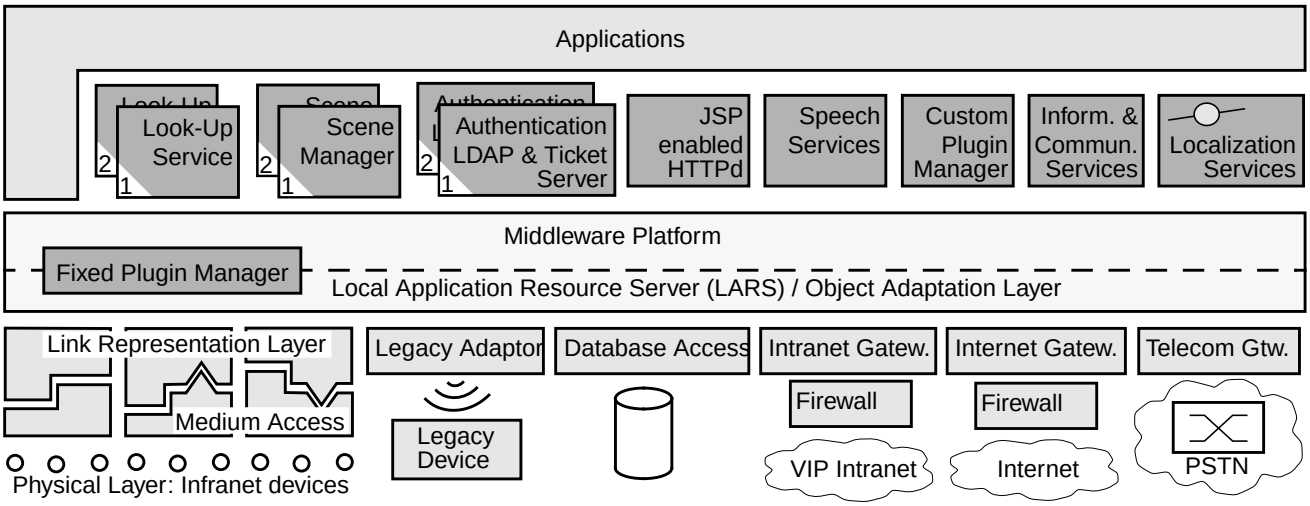

Figure 3 Software Architecture Overview for the Internet – Intranet – Infranet Integration Platform

might have memorized the login data during the session for Lara's convenience, it automatically asks for another ticket with access rights to meeting room control, thereby allowing the server application to keep these processes separate, even if the same operator is involved.

4.3. Scalability

The system is scalable from a variety of viewpoints.

The connection of more segments of infranet nodes requires additional cards in the interface servers, and additional servers if all card slots are occupied, thereby also distributing the infranet traffic. The performance of the control servers is sufficient for large office buildings.

Further types of infranets can be easily integrated, and legacy devices can be supported.

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.

The system is also down-scalable to one single computer for small, non-critical environments.

4.4. Software Architecture

The requirements discussed above lead to an architecture as depicted in Figure 3. The interconnecting element among all components is a CORBA based object oriented middleware, providing an abstraction of the physical distribution and network configuration. Main components of the system can be identified as follows.

4.4.1. Look-Up Service. 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. The Ticket Server ahead examines the access rights to the objects, thereby preventing unauthorized use.

This component is duplicated as primary and secondary server for redundancy reasons. When one server is dysfunctional, other parallel running servers can take over the functions.

Searching for devices or functions, the required properties are passed as templates 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".

4.4.2. Authentication, LDAP and Ticket Server. Initially, the 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. This component is also duplicated.

4.4.3. Scene Manager. The Scene Manager manages, maintains and stores scenarios. These scenarios are maintained in object oriented descriptions. This component is also duplicated.

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. Further, time-dependent animations can be defined this way, as far as the actuators (lamps, displays) allow.

There are two possibilities 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.

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

In order to keep processes on the lowest possible level for the most performant execution, previously known routines within the same sub-net are implemented as *fixed plug-ins*. 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.

A future extension could be a management component which examines custom rules and evaluates whether they could also be delegated to the lower level.

Plug-Ins are also used to define closed control circuits, e.g. temperature guided heating control, which are independent from access right restrictions. Their life cycle is managed by the Plug-In Manager.

4.4.5. Infrastructure Network Communication Layer. 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. This logical tier is harmonized in the Link Representation Layer and provides object oriented interfaces towards the middleware platform.

Some proprietary sub-systems within the building already provide data in abstracted form. In such cases, only the Link Representation Layer had to be implemented. An example is the heating and air conditioning system, which hides temperature sensors, internal control and climate actuators completely, providing “points of information” for data exchange.

Legacy adaptors provide access to devices which are not networked at all, but provide a remote control interface. A typical example are consumer electronics with IR control. The chips used for these control links are nowadays configurable to the protocol of a specific vendor via software.

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, and a unified interface is provided.

4.4.6. Local Application Resource Server (LARS). The 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 (user) applications, restricted by the respective access rights, which are managed using tickets.

4.4.7. Database Access. The database access component provides an object oriented view on relational databases. The underlying databases, which are accessed via JDBC [32], are shared with other services (employee, room and event information; room booking; communication services), thus allowing seamless integration of building control functions with existing or future building wide information and communication systems.

4.4.8. JSP Enabled HTTP Daemon. The Java Server Pages (JSP) enabled HTTP daemon provides the interface for simple control functions, and for external users (outside the building intranet), who are restricted by the firewall to use of port 80 communication. Java Server Page technology provides a convenient method to integrate dynamic content into the HTML pages.

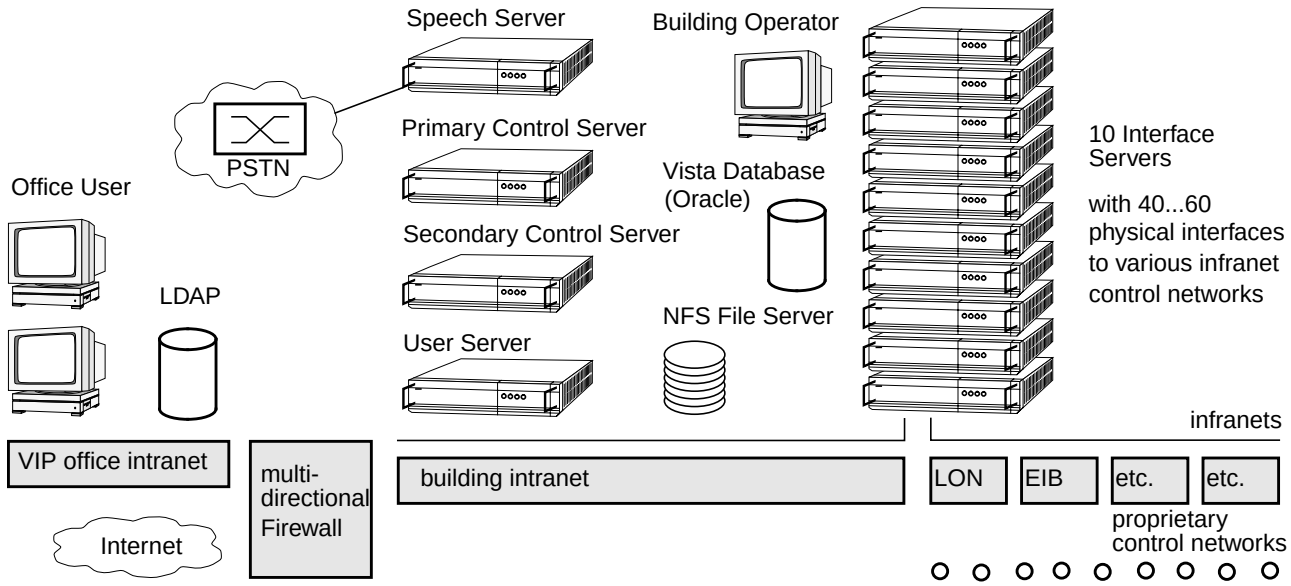

Figure 4 Hardware Architecture Overview

The widely used, reliable Apache HTTP server is used. It is combined with the TomCat JSP/Servlet engine [22].

4.4.9. Information, Communication and Localization Services. These components are used for the "VistaSite" location based communication, information and guidance system [34], which is beyond the scope of this paper.

4.4.10. Applications. 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 be connected with specifically implemented custom plug-ins.

4.4.11. Management. 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 [18][19][20], making this information also accessible by a centralized network management solution of the building.

4.5. Hardware Architecture

Based on the software architecture as described in section 4.4, the packages are installed on a hardware base – depicted in Figure 4 – as follows.

- Primary Control Server:
  - Primary Look-Up Service
  - Primary Scene Manager
  - Primary Authentication, LDAP and Ticket Server
- Secondary Control Server, with the respective secondary software packages
- User Server:
  - JSP enabled HTTP daemon
  - Custom Plug-In Manager
  - Middleware component for custom database connectivity
  - Interface packages for proprietary sub-systems
- Interface Servers:
  - Local Application Resource Server (LARS)
  - LON and EIB communication modules
  - Fixed Plug-In manager with Plug-Ins

5. 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. Some more examples from our labs are discussed in the outlook, section 6.

5.1. Speech and WAP Interface

In the described system, these interfaces are implemented for experimental and demonstrational use, only. The reliability of a speech activated control highly depends on the recognition engine, where we identified two major trends. First, there are consumer-level products, which are still very speaker dependent and depend on good microphones. Second, there are expensive solutions with hardware DSP support, available in optimized versions for telephony quality. However, the latter solution was out of the budget of the real system.

5.2. Graphical User Interfaces

Control applications are divided into

- such limited to firewall-permitted HTTP port 80 access, using only dynamically generated HTML-pages, optionally with embedded Java applets; for office control,

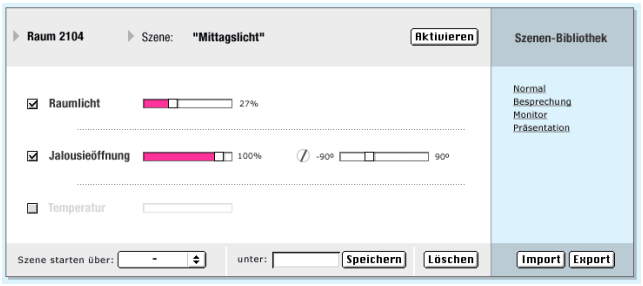

Figure 5 Lightweight office control applet

- rich and complex applications with full access to the middleware and the supporting components, for controlling complex technical systems such as event halls.

5.2.1. Lightweight Office Control. Employing any web browser (or alternatively small dedicated applications communicating via HTTP port 80), 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. Authentication can be host based or user based. Figure 5 shows a typical control applet.

5.2.2. Complex Control Applications. A main goal of the 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 profiles describing device configurations for a specific room in XML can be downloaded and sent to the design office. The creative people develop lighting scenarios (while sitting in the park) 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 6). 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, keeping the high-level traffic low.

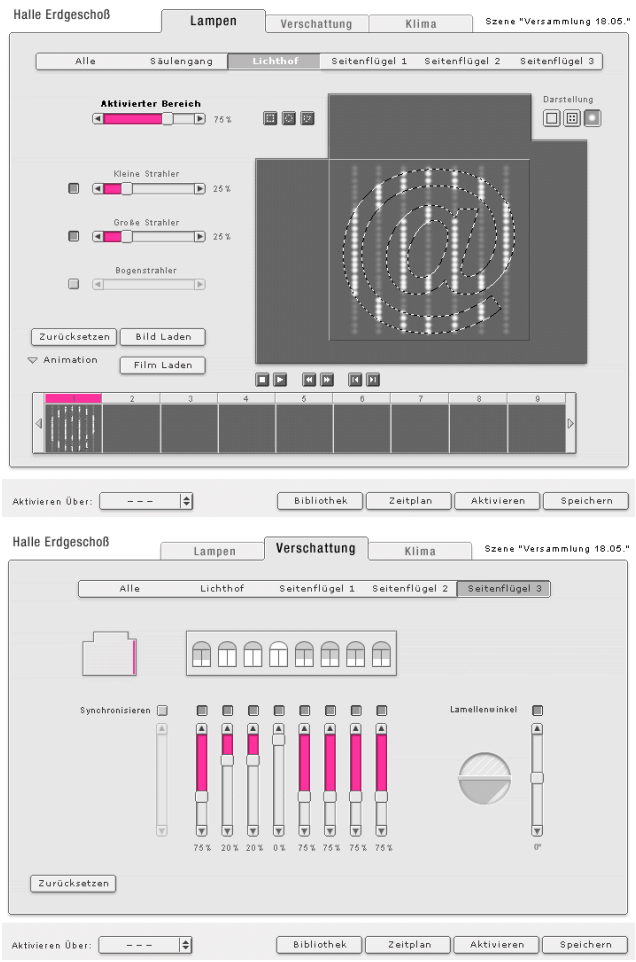

Figure 6 Graphical control applets

These activation commands can be sent 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.

6. Summary and Outlook

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. Some examples are:

In location-aware access control, restriction and surveillance scenarios, time and person dependent access rules process active badge information [28][29] and trigger door controls. Authorized persons receive paging information

(messaging service) or get their phone calls re-routed (communication services) [33]. Security rules can combine database knowledge about privileges: Equipment (with infrared badges and motion sensors) leaving a room together with an unauthorized person or even 'alone' can trigger alarms and close doors, while the authorized person can walk around with it.

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, before failure, only one person is necessary to change it. The room usage of the ad-hoc meeting is billed to the project of the responsible person.

Relying on planned activities and statistical information, energy consumption can be calculated and the most economical rate is negotiated (Mobile Agents) with the utilities, postponable processes can be re-scheduled.

7. References

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

[2] Estrin, Deborah; Govindan, Ramesh; Heidemann, John: Embedding the Internet. - in: 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] LonWorks Engineering Bulletin. - Echelon: Palo Alto, 1995

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

[21] BOFH: http://www.bofh.net

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

[23] 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. ISBN 3-540-41093-7

[24] Schmidt, Albrecht; Gellersen, Hans-W.; Beigl, Michael; Frick, Oliver: Entwicklung von WAP-Anwendungen. [Development of WAP Applications]. - in: Killat, U.; Lamertsdorf, W. (Eds.): Proc. of Kommunikation in Verteilten Systemen, KiVS, Hamburg, Germany, Feb. 20-23, 2001. - Berlin, Heidelberg, New York: Springer, 2001. ISBN 3-540-41645-5

[25] Link, Carsten; Luttenberger, Norbert: Sicheres Nomadic Computing in Intranet-Umgebungen – Problemstellungen und Lösungskonzepte. [Secure Nomadic Computing in Intranet Environments – Problems and Solutions] - in: Killat, U.; Lamertsdorf, W. (Eds.): Proc. of Kommunikation in Verteilten Systemen, KiVS, Hamburg, Germany, Feb. 20-23, 2001. - Berlin, Heidelberg, New York: Springer, 2001. ISBN 3-540-41645-5

[26] Pfeifer, Tom: Internet - Intranet - Infranet: A Modular Integrating Architecture. - in: Proc. of. 7th IEEE Workshop on Future Trends of Distributed Computing Systems, FTDCS'99, Cape Town, South Africa, December 20-22, 1999; Los Alamitos: IEEE Computer Society Press, ISBN 0-7695-0468-X

[27] DiGirolamo, J.A.: The VESA Home Network. A White Paper. - in: Proc. of the Home Networking 11053, London, 14-15 Sep. 1999

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

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

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

[31] 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

[32] Micklei, Andreas: Managing Relational Databases in Java. - Berlin: Technical University, 1999. Diploma thesis.

[33] Popescu-Zeletin, Radu; Pfeifer, Tom: A Modular Location-Aware Service and Application Platform. - Proc. of The Fourth IEEE Symposium on Computers and Communications, ISCC'99, Red Sea, Egypt, July 6-8, 1999

[34] Pfeifer, T.; Magedanz, T.; Hübener, St.: Mobile Guide – Location-Aware Applications from the Lab to the Market. - in: Plagemann, Th.; Goebel, V. (Eds.): Proc. 5th Int. Workshop on Interactive Distributed Multimedia Systems and Telecommunication Service, IDMS'98, Oslo, Norway, Sep 8-11, 1998, Springer: Lecture notes in computer science, Vol. 1483, Berlin et al., ISBN 3-540-64955-7, pp. 15-28