Intelligent Learning Systems, Adaptive Building Intelligence, ABI Mark II

a new ABI System built on the Open Services Gateway initiative

HSR
UNIVERSITY OF APPLIED SCIENCES
RAPPERSWIL

COMPUTER SCIENCE

Stephan Kei Nufer, Mathias Buehlmann, Joseph Joller, Rodney Douglas

snufer@ini.phys.ethz.ch, mbuehlma@ini.phys.ethz.ch

University of Applied Sciences, Rapperswil and Institute of Neuroinformatics, University / ETH Zürich

uni | eth | zürich

I. Introduction

With the advent of programmable systems, building managers must gradually adjust lighting levels in response to occupant requests and thus definitely opens the possibility for an **Intelligent Building (IB)** to handle such tasks by considering aging of devices, changing sky and user behaviors as well as taking possible structural variations into account. An IB can lower the energy consumption as well as maximizing visual user comfort and also lessens cost-intensive maintenance that elsewise would be necessary.

In our term project (Intelligent, Learning System built on the Open Services Gateway initiative), a new ABI (Adaptive Building Intelligence) System has been developed that was based on OSGi. OSGi has emerged into a very powerful and adaptive system framework that enables an entirely new category of smart devices due to its flexible and managed deployment of services to be implemented and integrated. With the usage of such a framework we achieved a stable, maintainable and an expandable system.

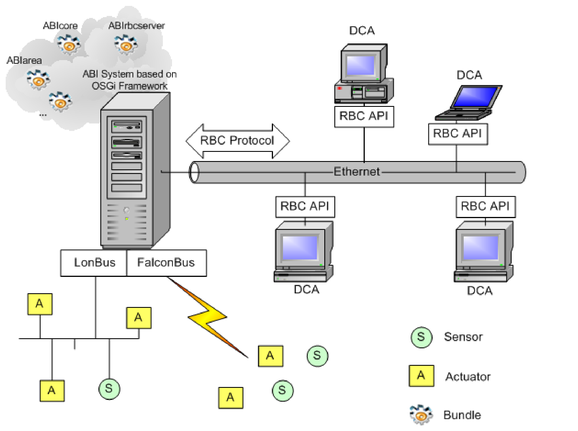

Figure 1: *ABI System Architecture*

II. System Architecture

In this diploma thesis we introduce a new multi-agent approach that can be deployed and used to control a commercial building equipped with casual sensors and effectors. The communication between the agents is hereby described by a specifically developed protocol called RBC (Remote Building Control). It has been proven that the new architecture simplifies integration testing, debugging and maintenance remarkably.
In addition to the overall infrastructure, we studied several possibilities of machine learning algorithms that can be applied in an ambient working and living environment.

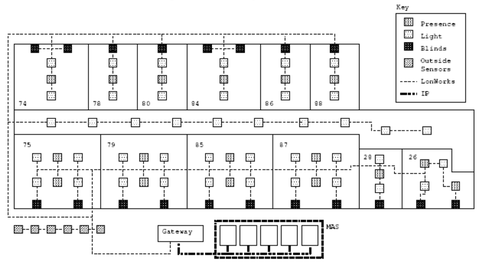

Figure 2: *Multi-sensor Environment*

In order for any control system to interact with a real working and living environment, access to different indoor as well as outdoor sensors and effectors must be provided. It this diploma thesis we incorporated new sensors and effectors that are addressed by a dedicated fieldbus network, LonWorks.

RBC Protocol

The purpose of the RBC Protocol is to describe the protocol to be used for exchanging information between the ABI System (RBC Server Bundle) and custom application programs (Agents) that need guaranteed reliable transmission of data in a simple, ascii-based protocol. One major use of this protocol is to enable agents to retrieve changing device information and on the other hand commands which are executed in the RBC Server similar to remote procedure calls(RPC). Hereby it provides a standard that all ABI Agent applications need to adhere when communicating to the ABI System.

RBC API

The RBC API specifies a new compact application layer standard which exemplary has been implemented by a concrete Remote API that implements the RBC Protocol requirements as it was prescribed by its specification. This concrete implementation remotely communicates to the ABI System Server and herewith provides an ideal basis for distributed agents such as Device, User Interaction or Logging Agents to use for their realization.

The benefit of such a system is that all distributed Agents such as the Area Controller Agent can now be developed independently without having to deal with the ABI System itself. Thus improving any development practices such as complex integration testing, debugging and maintenance remarkably.

III. Distributed Agent Applications

In order to test the hot plugging capabilities we developed a variety of distributed agents that will either help to test the overall infrastructure or will provide some advanced management utility functions that can be used to structure the entire building within its interconnected network of devices. It should be clear that by such a huge environment with lots of occupants in it we need to consider a convenient technique to facilitate the usability of such applications. Not only the usability of the agents are meant hereby but also to achieve a low effort in installing and configuring the agents. One might wonder why such agents are needed in the first place. The answer is quite simple. Some of the workstations within a larger space might actually rather correspond to a lab then a regular office and hence suffers from directly accessing available effectors such as lights or blinds. Hence by providing a limited view into such an environment we can easily potentiate the manual control over those devices with web based agent application. Web-based since they will be available through Java Web Start.

Area Controller Agent

This user interaction agent allows any environmental part to be accessed in a distributed kind of fashion. Task such as: Lifting up desired window blinds, switching on and off the lights and a whole bunch of other features are supported herewith

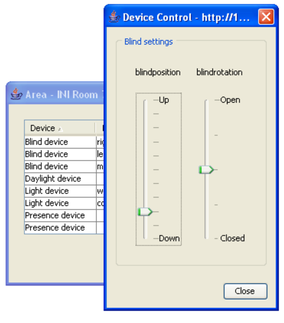

Figure 3: *Area Controller Agent. Moving blinds...*

Area Controller Agent PC Presence

This user interaction agent combos the latter listed agent with an additional Remote PC Presence detector software piece. This sensor serves as an additional personal presence detector that tracks down the users mouse motions and keyboard hits. The reason of its development is the urgent need of additional presence detection due to the fact that each presence sensor we are currently working with, rather correspond to regular movement detectors which are only capable of detecting movements of persons within a certain range. Hence when movements are getting sparse or when the sensors simply don't have direct intervisibility to occupants, regular PIR sensors start to fail to correctly detect the presence of an environment. To counteract this issue so called *PC Presence detectors* have been developed that help to correctly sense presence of a room. In order to enhance the deployment of the PC Presence detectors, we let them run as thin client applications, right underneath the regular Area Controller Agent.

Area Admin Agent

The Area Admin Agent stands for ABI system administrator agent and primary serves as a global management application with the purpose to provide a solution to centralize ABI system security. Tasks such as supervising rooms, make new device assignments or creating new areas and devices are ordinary jobs done with this user interaction agent. Since it provides global access rights it possesses the authorization to perform any actions that are possible within the range of control.

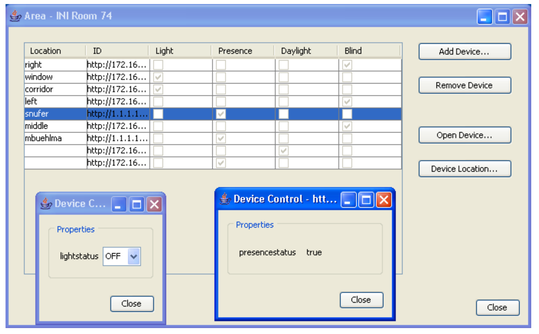

Figure 4: *Creating new Areas*

Area Logger Agent

Area Logger Agent is a logger agent that records all sensory and effector as well as environmental structure changes into a relational database that can later be useful to employ data mining using real live data from the INI.

IV. Device Agents

Each space that is controlled by an IB contains of a set of Device Agents. Hereby each space is controlled using a novel IBF (Intelligent Building Framework) that among other things introduces a new algorithm that considers the changing strengths and weaknesses of different prediction algorithms. Each prediction algorithm hereby tries to learn its dynamic living and working environment and within its occupants in a self-adapting way. In regard to building intelligence, the task of the IBF is then to compete them against each other and will eventually filter out those algorithms which were rather wrong in their prediction and boost algorithms that rather tended to be right. This is reasonable since each environment is different. - And likewise each algorithm.