

Smart Documents for Home Automation

Andres Albanese

Via F. Busoni 17
50019 Sesto Fiorentino (FI), Italy
myhomevillage.com, aalbanese@ieee.org

Abstract. This work describes the use of well known computer applications to enable smart home users to monitor and control their homes using customized documents. Middleware written in applescript and perl-cgi was used to integrate the computer applications with the OpenWebNet protocol used in home automation. The events triggered by the applications are easily log by web and mail servers to facilitate diagnostic operations and their archival. This software was tested on the implementation of “The Smart Door Project” to remotely manage door access, monitor the door and archive the events. One of the features is that the door opens after the user receives an e-mail with the magic words “Apriti Sesamo” in the subject field, and “Alibaba” in the text.

Keywords: Applescript, Domotica, Home Automation, Middleware, OpenWebNet, System Integration, Smart Door, Smart Home, Wiki Server.

1 Introduction

With the evolution of home automation, there is an increase in complexity due to the large number of systems, modules, sensors, scenarios, displays, options, and features. Ideally, commands are simple clicks; but in smart homes, there is a large variety of commands, ranging from physical keys in walls close to appliances, to remote virtual keys and gestures in mobile devices, plus many scenarios triggered by a variety of events. Therefore, there is a need for a well maintained documentation containing a detail list of instructions for the house on what to do during the opening or closing operations, when going to sleep or awakening, while enjoying the home theater, etc. These instructions must deal with the settings of the zones (rooms and areas) and systems (lighting, irrigation, climate control, sound distribution, and burglar alarm). In addition, this instructions should be adapted for different users (kids, adults, and seniors).

Every manufacturer provides user manuals and a remote control for easy operation of their equipment; but it is possible to end up with many manuals and remote controllers, at least one for each brand of equipment in the several areas of the home. Also, users get information from the web by googling for solutions to frequent problems; therefore, there is a need for an integration between the information and the functionality of the home devices. Today, most automated homes have one or more computers connected to the internet; and, it is possible to use wiki[1] applications to

save, manage and search specific home information like documents, inventory of the components, list of service providers, maintenance and operating instructions, detailed pictures, glossary and index for easy reach; but, a software middleware [2] is needed to make all the home automation modules in the house available as objects to be inserted and used by applications in the computer.

It is the purpose of this work to describe how to use common applications, to control a home by interacting with documents provided by installers and modified by users. For example, the search for a received e-mail containing the instructions for opening the house enables the user to modify it and resend it to her/himself; with objective that the computer executes the instruction list after receiving the e-mail.

2 Smart Documents

The following subsections describe the applications tested in the creation of smart documents for home automation.

2.1 Mail. Using Mail Rules for Automations

Most digital documents allow users to send an e-mail with preset fields with a single click. In our case we send an e-mail with the Subject “Apriti Sesamo”, and a body message containing the “magic” word “Alibaba”. The receiving Mac computer at home filters the received e-mail, using the Rules features in the Mail application, and runs an applescript after checking all fields (sender, recipient, subject, and magic word in content.) Below, there is an example of a 2-line applescript to open the door by closing a contact for 0.5 sec, using an OpenWebNet¹ command sent to the home controller and playing a message saying “the terrace door is OPEN”:

```
do shell script "echo \"*1*18*72##\" | nc 10.0.0.82 20000"
tell current application to say "The terrace door is OPEN"
```

If you would like to try the applescript above, it will have to be customized for the switch OpenWebNet command (*1*18*72##), and the IP address of the My Home BTicino² gateway. Most e-mail services today support secure SSL to encrypt messages, in this case the text Alibaba; and they move selected e-mails into folders for later retrieval.

2.2 iCal. Calendar Events and Reminders to Schedule Automations

It is possible to have repetitive events and reminders in iCal³ (Apple’s calendar application). Each event and reminder entry have one or more alarm fields to specify a sound, send e-mails and/or run a script at a specific time and date. The iCal application can be used to track our scheduled appointments organized by categories,

¹ OpenWebNet is a protocol standard for home automation.

² My Home BTicino is a home automation product of Legrand Group.

³ OS X Server, Mac OS X, MacBook, iMac, Preview, Pages, Numbers, Keynotes, Mail, iCal iTunes, iBooks, iBooks Author, Applescript, Automator, Mac, iPad, iPod and iPhone are Trademarks of Apple Inc.

therefore, home automation commands can be scheduled in a category. For example, on our anniversary, an event lights up the color LEDs in the hall and play the romantic Playlist in iTunes. The same process is possible for the summer irrigation and climate control.

The server has a middleware consisting of applescripts used by the events and reminders to execute automations as alarms. Below, there is a 2-line applescript to schedule a repetitive command using iCal in Mac OS X:

```
do shell script "echo \"*1*16*57##\" | nc 10.0.0.82 20000"
Tell current application to say "irrigation ON for 15 minutes"
```

If you would like to try the applescript above, it will have to be customized for the switch OpenWebNet command (*1*16*57##), and the IP address of the My Home gateway. Use the AppleScript Editor to help with the creation of the applescript. Save and name your applescript with a name like irrigate.app and close the editor to start the iCal application. Edit the iCal event fields: time, repeat, country time zone, etc. One of the alarm fields should contain the applescript irrigate.app. Other alarm fields may be used to send e-mails, warning signals, etc ... before or after the event. We can even use an alarm field to start iTunes with the music preferred by a user during the irrigation time.

2.3 Wiki. A User Web Interface for Home Automation

Most house computers come with a web server application included to service web requests from the local area network and Internet; that, users can access through a variety of browsers. The Wiki Server is a type of web server used for collaboration that enables users to create and publish their own wikis, blogs and podcasts. These web services are useful to communicate and collaborate with friends, co-workers, and family members. The wiki server manages the access to the information contained in the wiki pages and shows documents according to user and group permissions, and their enabled services: calendar and blog. Users are required to login in before accessing the wiki pages. The wiki pages display the specific commands, web cam images, and text and photographs.

A wiki document may contain live images from network cameras, making it easy to check a room, to see if the lights and shutters are working fine. Family members and some friends may have access to external views of the home, to get an indication that the exterior lighting and irrigation system are properly working, while the owners are on vacation. The server enables the use of virtual private networks (VPN) to allow mobile devices to access home information remotely as if they would be in the local network at home. The Wiki application is useful for people that need to delegate the monitoring of a patient remotely while limiting the devices controlled by the delegated person. Each wiki has a folder containing a limited number of .cgi files with commands for the home controllers.

Mail, iBook, iTunes, Keynotes, Numbers, Pages, and Preview documents can contain hyperlinks to .cgi files in a server; enabling the user to control her/his home

using a simple click while reading the document, playing a presentation, or even listening to music! A click on a URL sends a request to the HTML server to execute a .cgi file. The CGI-Executables folder in the server contains a list of possible .cgi files with scripts containing command lists to control home switches.

3 Smart Door Implementation

There are several automations to open a door remotely, and to turn ON a light when a person approaches a front door. However, in our implementation in Florence, Tuscany, there are synergies among the automation systems, the network camera, the computer at home, the e-mail application, and the motorized lock. The IR sensor triggers the automation to welcome visitors by turning ON the light and ringing the front door bell. When the light goes ON, the network cameras sends an e-mail with the image of the area around the door. The home owner receives the e-mail and she/he forwards the message to her/himself after adding the keyword, in this case “Apriti Sesamo”. The computer activates the motorized lock using an OpenWebCommand sent by an e-mail Rule after checking the sender, subject and content fields of the received e-mail. All the e-mail involved in the process are saved and archived.

References

1. Boulos, M., Maramba, I., Wheeler, S.: Wikis, blogs and podcasts: a new generation of Web-based tools for virtual collaborative clinical practice and education. *BMC Medical Education* 2006 6(41) (August 15, 2006)
2. Bauer, F.L., Bolliet, L., Helms, H.J.: Software Engineering. In: Naur, P., Randell, B. (eds.), Garmisch, Germany, Brussels, Scientific Affairs Division, October 7-11, 1968. NATO (1969)