Implementation of Voice Control Interface for Smart Home Automation System

Sunčica Milivojša, Sandra Ivanović, Tatjana Erić
Faculty of Technical Sciences
University of Novi Sad
Serbia

Marija Antić, Nikola Smiljković
RT-RK Institute
Novi Sad, Serbia
marija.antic@rt-rk.com

***Abstract*— In this paper, we implement a voice command interface for smart home automation systems. The proposed interface defines command patterns, which can be easily mapped to specific devices, device functionalities and device locations. Also, the control of devices in a group is supported. The proposed voice command interface is implemented in way that supports various voice recognition engines.**

Keywords—voice command, home automation

I. INTRODUCTION

Home automation (HA) has become very popular and widespread over the past few years. Various sensors and smart devices are already available on the market, as well as HA systems, which let users connect, monitor and control their devices. Most of the existing HA systems are controlled by mobile or web applications. Meanwhile, we are witnessing the growth of popularity of voice command interfaces (VCI). Recently, a number of devices have appeared on the market, which are powered by the online voice recognition engines, and can serve as the VCI for HA systems [1], [2]. Mechanisms are provided for developers, to create custom HA web services, which respond to voice commands issued by the users. Some of the vendors even provide APIs, which are specifically targeting HA devices [3]. However, these APIs are still developing, and the number of possible commands is limited. Also, when using such an API, developers are required to host their HA web service with the vendor providing the API. On the other hand, using the general purpose API by the same vendor provides us with more freedom when deploying the HA web service responsible for fulfilling voice commands [4].

Our goal is to build the VCI for the already existing HA system [5]-[7]. We want our VCI to be compatible with the existing voice recognition services [1], [2], as well as the custom ones.

According to [5], every smart home device is characterized by the set of services it supports, such as light service, outlet service, etc. Device service describes a group of device functionalities. In this paper, we implement the VCI for smart HA, based on device services. This VCI can be used with the arbitrary voice recognition service. In Section II, the overview of the proposed VCI is presented. Section III describes two possible implementation of the VCI. The proposed VCI is confirmed by testing it with two major voice recognition engines [1], [4].

This work was partially supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia, under grant number TR32030.

II. DESIGN OF VOICE COMMAND INTERFACE

In this section, we briefly introduce the proposed voice command interface for smart HA. First, we describe the way devices are represented within the HA system. Then, we describe some of the supported command patterns.

HA system consists of endpoints (devices) and a central gateway unit [6]. Endpoints are divided into two groups – sensors and actuators. States of sensor devices can be monitored, but they cannot be changed by the user. On the other hand, users can control various states of actuator devices. Depending on their functionality, devices register one or more different services. Service describes the functionality of the device [5]-[7]. Example of a sensor service is the *temperature service*. The user can only check the temperature of a device. Example of the actuator service is *light service.* Properties of this service can be changed by the user, and the state of a physical device changes accordingly.

We will design the commands to control or read properties of individual devices, as well as to control groups of devices. In the text that follows, we introduce some of the supported command patterns. Due to limited space, we introduce only the basic ones, although the supported commands set is larger.

A. Command patterns targeting individual devices

In every command targeting an individual device, we can identify two main parts: *action* and *device name*. The first part of the command can be mapped to the targeted device service. The rest of the command will be mapped to the name assigned to the device within the HA system.

Let us consider the command *turn on light*. In this case, *turn on* is the part of the command corresponding to the desired HA system action. The utterance *light* represents the device part of the command. In order to correctly execute the command, the device with the name *light* must exist in the targeted HA system, and it must support the service targeted by the command *turn on*.

The problem we are facing is the fact that some voice recognition engines require that all possible command patterns are defined in advance, unless the API for HA is used [2], [4]. As already said, using the HA API limits our web service

978-1-5090-4014-8/17/$31.00 ©2017 IEEE

deployment options. So, we choose to use the general purpose API by the same vendor [3]. This means that all possible device names should be included in the patterns we define in our VCI. Therefore, we extend the supported command patterns to support *location descriptors*. Location descriptors can be of the following types: position of the device inside the room, name of the room, or name of the floor. The variable number of location descriptors can be combined, and used as part of device name. For example, device names can be *table light*, *kitchen light*, *study table light*, etc. This allows us to build a scalable VCI, supporting the variable command length and multiple device names

Commands for sensors are similar, but can be used only to read the value of device properties. Depending on the sensor type, this information can be the value of the temperature or humidity, value of the contact state (opened or closed), flood or smoke sensor state, etc.

B. *Command patterns targeting groups of devices*

Controlling individual devices is not very practical, as it eventually results in the need for a user to remember long device names. Also, to control multiple devices, the user has to issue a number of commands in a row. This is acceptable if the user wants to control devices of different type, but it is preferable to control a group of devices supporting the same service with group commands. In Section II.A, we introduced rooms as possible *location descriptors.* Since the devices can be assigned to the rooms within the HA system [6], we can use room names to control groups of devices. Group commands for devices supporting the same service include a keyword *all*. This keyword is a signal to the HA system that the action should be performed on a group of devices sharing the same location. For example, the command *turn on all lights in the kitchen* can be split into the part defining the targeted service (*turn on*) and the part defining the location (*kitchen*).

The existing HA system [5]-[7] also offers users the possibility to configure action sets targeting groups of devices of arbitrary type. We can use the two-part voice command pattern described in Section II.A to execute these user-defined action sets. In *action* part of the command, we dedicate the keyword *run* to indicate that the execution of the user defined action set is expected. The other part of the command defines the action set name. For example, the command *run morning action*, executes a user-defined action set, memorized by the name *morning action*.

III. IMPLEMENTATION

The proposed VCI was implemented and tested by using both Amazon Alexa Voice [1], and Actions on Google services [2].

In case of [2], the command issued by the user is processed by Alexa cloud, based on the command interface defined in the custom Alexa skill [4] – Fig. 1. This service creates the structured input for the HA system, in which the action part of the command is distinguished from the device name or the device location. Based on the structured data obtained from Alexa service, the HA system executes the desired command.

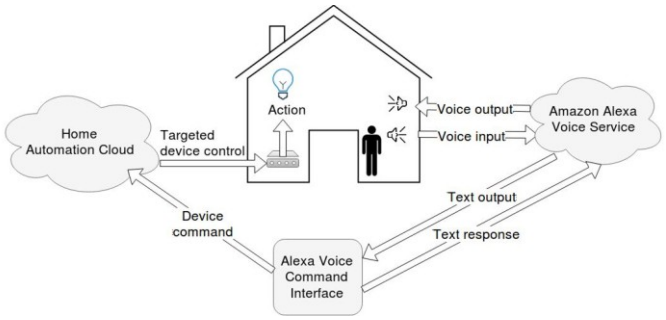

Fig. 1 – Architecture overview of the HA system with VCI

On the other hand, when implementing the proposed VCI with [1], we could only obtain the unstructured data (text) from the voice recognition service. In order to create the structured input for the HA web service, fulfilling user commands, we needed to implement the command parser component within our web service. This component was implemented using ANTLR command parser [9], which was configured to use the same grammar as Alexa skill. The parser component produces the structured output, similar to the one created by [4]. This structured data was used as the input for the same HA web service, as in case of [4].

Functionality of both implementations was confirmed by issuing various voice commands, and confirming that the desired action was executed within the existing HA system.

IV. CONCLUSION

The structured and scalable VCI for HA was implemented using two major voice recognition services. By implementing the command parser component using ANTLR, we were able to use the same HA web service is to process commands coming from both of the voice recognition services. The functionality of the implemented VCI was confirmed within the existing HA system.

REFERENCES

[1] Alexa Voice Service, [Online] Available: https://developer.amazon.com/alexa-voice-service
[2] Actions on Google, [Online] Available: https://developers.google.com/actions/
[3] Smart Home Skill API Reference, [Online] Available: https://developer.amazon.com/public/solutions/alexa/alexa-skills-kit/docs/smart-home-skill-api-reference
[4] Understanding Custom Skills,[Online] Available: https://developer.amazon.com/public/solutions/alexa/alexa-skills-kit/overviews/understanding-custom-skills
[5] M. Savic, V. Mihic, I. Lazarevic and M. Sekulic, "Modular home automation software with uniform cross component interaction based on services", *Proc. of IEEE 2015 ICCE – Berlin*, 2015.
[6] M. Tucic, V. Moravcevic, G. Velikic, Đ. Saric and V. Mihic, "Device abstraction and virtualization: Concept of device in device", *Proc. of IEEE 2015 ICCE – Berlin*, 2015.
[7] M. Tucic, V. Moravcevic, R. Pavlovic and A. Majdak, "An Approach for Uniform Representation and Control of ZigBee Devices in Home Automation Software", *Proc. of IEEE 2015 ICCE – Berlin*, 2015.
[8] C. Vimala, V. Radha, "A Review on Speech Recognition Challenges and Approaches", *World of Computer Science & Information Technology Journal*, vol. 2, no. 1, pp. 1-7, 2012.
[9] T.Parr, *The Definitive ANTLR 4 Reference*, Pragmatic Bookshelf, 2013