www.cr.rwth-aachen.de
cr@ip.rwth-aachen.de

Editors:
Univ.-Prof. Dr.-techn. Sigrid Brell-Cokcan
Dr.-Ing. Thomas Adams

RWTH Aachen University
Chair of Individualized Production
Campus-Boulevard 30
52074 Aachen
Germany
www.ip.rwth-aachen.de

Editorial Assistance:
David Lukert B.Sc.

Original cover:
Lukas Kirner M.Sc.

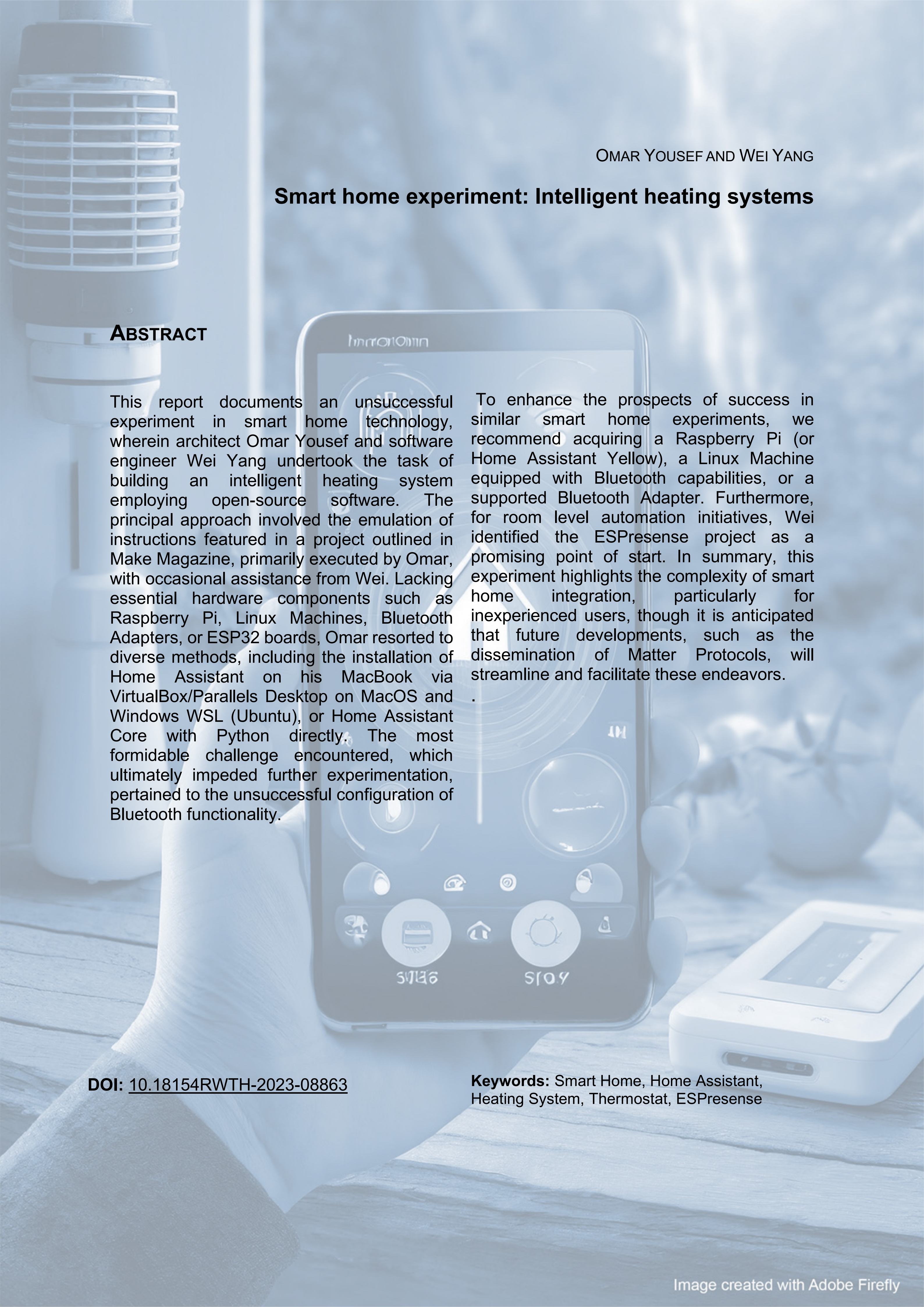
Funded by BLE Seed Fund of Faculty 2 - Grant of RWTH Aachen University

The content of the book was created for teaching purposes as part of the Research Driven Projects course.

RWTH Aachen University
Templergraben 55
52066 Aachen

First published 2024

Available via the institutional repository of RWTH Aachen University:
DOI: 10.18154/RWTH-2023-08854



OMAR YOUSEF AND WEI YANG

Smart home experiment: Intelligent heating systems

ABSTRACT

This report documents an unsuccessful experiment in smart home technology, wherein architect Omar Yousef and software engineer Wei Yang undertook the task of building an intelligent heating system employing open-source software. The principal approach involved the emulation of instructions featured in a project outlined in Make Magazine, primarily executed by Omar, with occasional assistance from Wei. Lacking essential hardware components such as Raspberry Pi, Linux Machines, Bluetooth Adapters, or ESP32 boards, Omar resorted to diverse methods, including the installation of Home Assistant on his MacBook via VirtualBox/Parallels Desktop on MacOS and Windows WSL (Ubuntu), or Home Assistant Core with Python directly. The most formidable challenge encountered, which ultimately impeded further experimentation, pertained to the unsuccessful configuration of Bluetooth functionality.

To enhance the prospects of success in similar smart home experiments, we recommend acquiring a Raspberry Pi (or Home Assistant Yellow), a Linux Machine equipped with Bluetooth capabilities, or a supported Bluetooth Adapter. Furthermore, for room level automation initiatives, Wei identified the ESPresense project as a promising point of start. In summary, this experiment highlights the complexity of smart home integration, particularly for inexperienced users, though it is anticipated that future developments, such as the dissemination of Matter Protocols, will streamline and facilitate these endeavors.

.

DOI: 10.18154RWTH-2023-08863

Keywords: Smart Home, Home Assistant, Heating System, Thermostat, ESPresense

Image created with Adobe Firefly

1. Introduction

1.1 Motivation

As energy cost becomes higher, we wanted to build an intelligent heating system for a single apartment. The system should be able to be controlled by change in the room temperature and the presence of people in it.

Omar is an architect without much knowledge of programming or IoT devices while Wei is a software engineer with programming experience in C++ and python and some knowledge of IoT devices. Both of us have little experience with smart home projects or productions.

We hope to get some hands-on experience with smart home devices and software and interdisciplinary communication and collaboration, which may also enlighten us on why digitalization in construction is such a difficulty.

1.2 Research Question

How difficult would it be to build an easy smart home heating system for unexperienced people without a computer science background?

There are already several IoT devices and some open-source platforms for smart home. How do general people (not experienced) feel to setup a smart home system?

With devices and services from big companies like Apple, Google, Amazon, one could control lights, heating systems, air conditioners, curtains, security cameras by Siri, Google Assistant and Amazon Alexa. Could smart home devices easily be setup and be controlled by these hubs or smart phones?

1.3 State of the Art

One of the simplest solutions could be Tuya TRV603 Wifi Thermostat[1]. TRV603 (about 35 € with free shipping at the time of the paper, while normal sale price is about 64 €) could be controlled by your phone via Smart Life App which also contains a beta feature to control the thermostat, depending on your location.

This solution might not give us much enlightenment about how the smart home system works, however, it could provide comparison baseline for our experiment.

2. Research Methods

2.1 Make Magazine Project Instruction

The reference used, which was also the most helpful, was the article *Heizung unter Kontrolle* (Heating Under Control)[2] from the Make Magazine. For this project a specific thermostat, Equiva Bluetooth Smart Radiator Thermostat (CC-RT-BLE-EQ) (Fig. 2), and a thermometer, Xiaomi Mijia Smart Thermometer Bluetooth (Fig. 3), were required. The next step would be installing Home Assistant on the PC. This was the very challenging part which required a lot of research and effort, because unfortunately it was not designed for the typical user, and it needs someone who is very familiar with computer science and some specific modern technology. The Home Assistant website has a guide to run the application on each of the operating systems, whether it was a Linux Machine, MacOS, Raspberry Pi, Windows, or even others. The experiment carried out by Omar was on a MacBook (Intel CPU model) while Wei did some more exploration on a Raspberry Pi 4. As we worked in different countries during the experiment, we could only collaborate online, which made the experiment even more challenging.

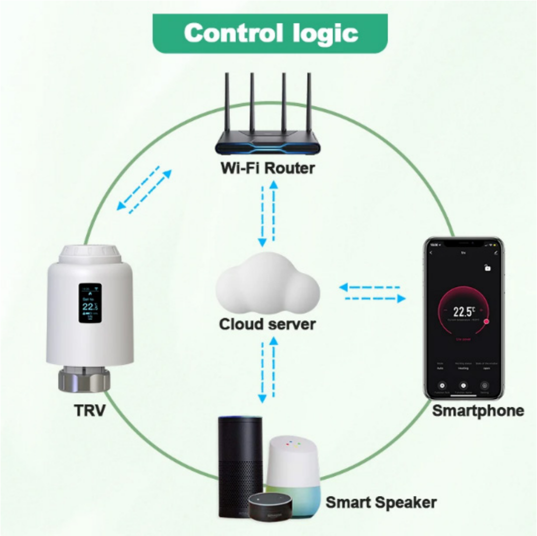


Fig. 1: Control Logic of TRV603 [1]

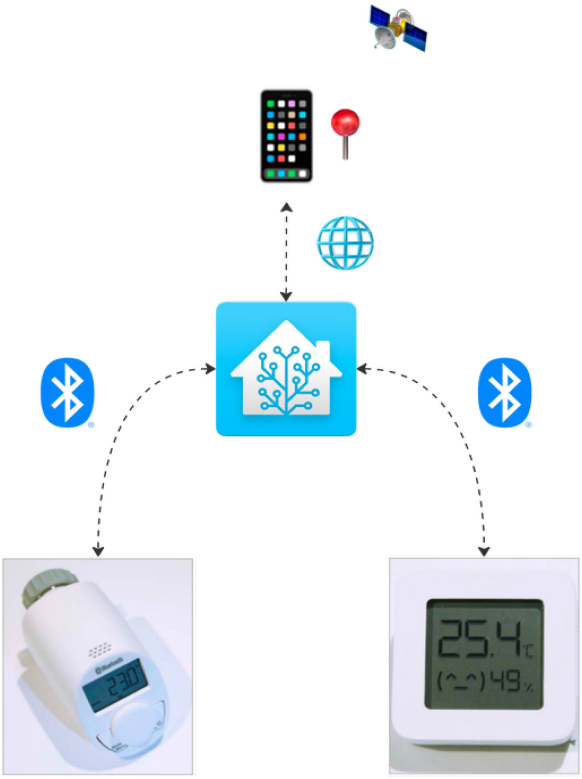


Fig. 4 Overview of the Heating System

2.2 Market Research

We wanted to check the possible products from Amazon (or AliExpress), compare their features, costs, and complexity, so that we could understand the state of art for smart home. The review from customers was also helpful for us to find current problems for smart home.

2.3 Paper Research

We wanted to check previous research status for smart home controlled heating system, their concerns and protential.

2.4 Open-Source Projects Research

As smart home devices are controlled by complex software, we hoped to find potential open-source projects from GitHub or GitLab. This might also give us some overview status for the smart home community.

3. Results

3.1 Architecture

It is basically the use and connection of the thermostat and the temperature sensor with Home Assistant, and this happens through WIFI and Bluetooth connections (as shown in Fig. 4).

3.2 Market Research Results

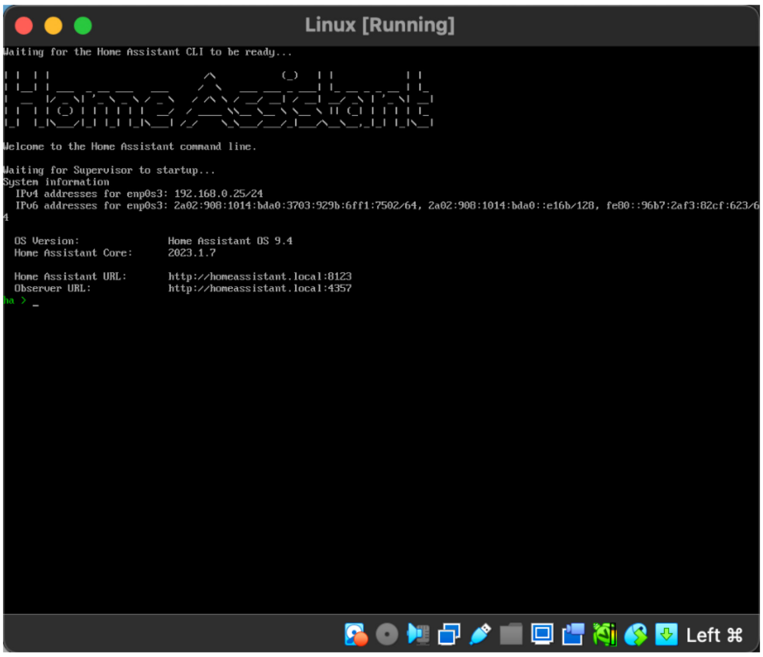
Most of these thermostats can be found on the German Amazon Store. These are some examples that could give an overview of the market regarding choosing the best thermostat. The last two options are not smart thermostats, but they are used in the table as a reference to show the difference between smart thermostats and normal ones.

Picture	Type	Cost	Method
	eQ-3 Heizkörperthermostat – basic HmIP-eTRV-B	59,99 €	via app MagentaHome App Pro Magenta SmartHome Pro
	Hama 00176592 Smart Radiator Thermostat	39,99 €	via app Alexa
	AVM FRITZ!DECT 301	58,00 €	via app
	Equiva Bluetooth Smart Radiator Thermostat	24,87 €	via app
	TP-Link Kasa Smart Radiator Thermostat	49,90 €	via app Kasa Smart App Alexa Google Assistant
	Heimeier K Thermostat Head	14,00 €	Manual
	Oventrop Thermostatkopf Uni-LH	17,99 €	Manual

Table 1: Thermostats in the market

3.3 Home Assistant Setup & Configuration

In the guideline of the Home Assistant page for MacOS[3], it states that another operating system should be used on the MacOS, which meant that a ‘virtual’ Linux operating system needed to be used to make the Home Assistant application compatible with the MacOS. This could be done by downloading an image called VirtualBox, where it enables the user to run a Linux on their MacOS. From a link on the website, VirtualBox can be downloaded, and all the steps are available on how to install it,configure the new virtual



```
Waiting for the Home Assistant CLI to be ready...

Welcome to the Home Assistant command line.

Waiting for Supervisor to startup...
System information
  IPv4 addresses for enp0s3: 192.168.0.25/24
  IPv6 addresses for enp0s3: 2a02:908:1014:bda0:3703:929b:6ff1:7502/64, 2a02:908:1014:bda0::e16b/128, fe80::96b7:2af3:82cf:623/6
4

  OS Version:               Home Assistant OS 9.4
  Home Assistant Core:      2023.1.7

  Home Assistant URL:       http://homeassistant.local:8123
  Observer URL:             http://homeassistant.local:4357
ha > _
```

Fig. 5 Home Assistant OS in VirtualBox

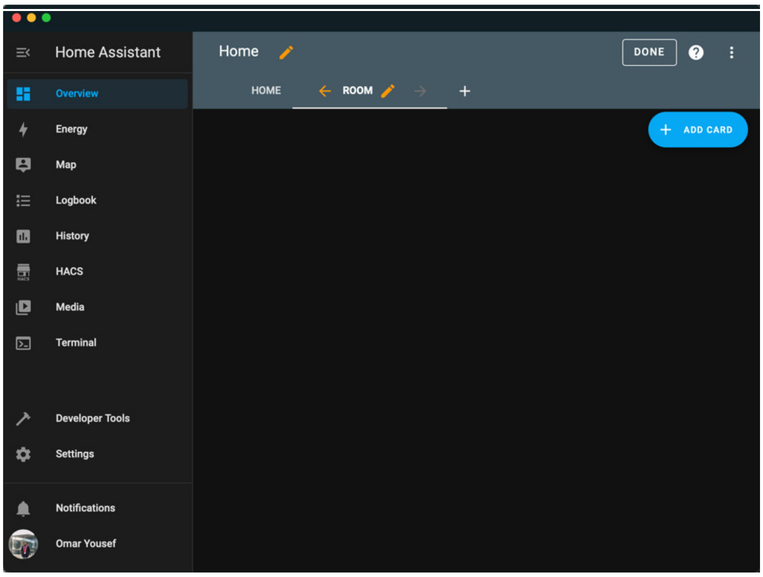


Fig. 6 Create Room in Home Assistant

machine, set it up and run it. Once the Linux virtual machine is up and running (Fig. 5), Home Assistant can then be normally run, with instructions on how it could be run, available as well on the website.

Home Assistant can then be accessed through the address (http://homeassistant.local:8123) in the browser or directly on through the companion application, simply installed from the App Store on the MacOS or iOS/Android. When Home Assistant is then started, the steps were followed from the Make Magazine article. The first step was creating the room which was experimented on Home Assistant, naming it, and optionally giving it a logo (Fig. 6).

Then, a very important step was to go to the settings and enable Advanced Mode, which then allowed the download of the installation of SSH & Web Terminal as one of the add-ons of many available on Home Assistant. It was a bit tricky to actually run it and make it work, since a very small detail was required to be known. Before it could be run, a password was required to be typed in the password line under ssh, via the Configuration tab, and then the add-on was launched (Fig. 7).

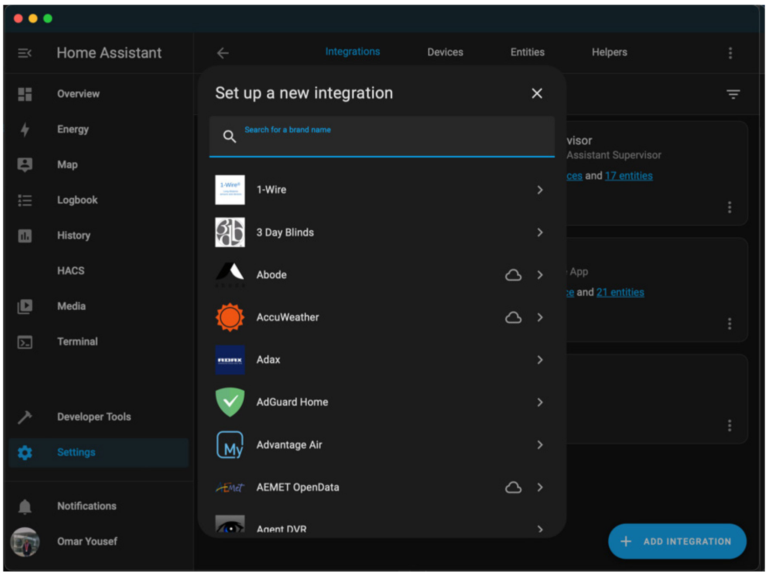
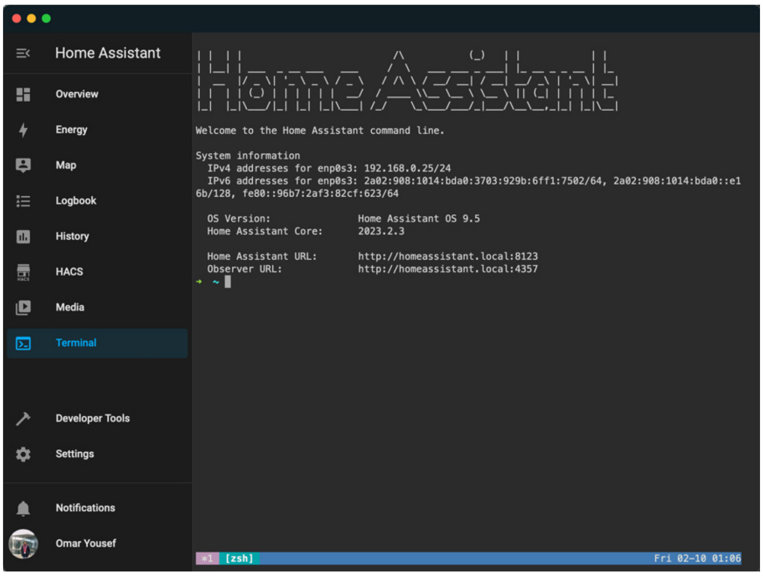


Fig. 7 SSH & Web Terminal Addon for Home Assistant



```
Welcome to the Home Assistant command line.

System information
  IPv4 addresses for enp0s3: 192.168.0.25/24
  IPv6 addresses for enp0s3: 2a02:908:1014:bda0:3703:929b:6ff1:7502/64, 2a02:908:1014:bda0::e1
6b/128, fe80::96b7:2af3:82cf:623/64

  OS Version:               Home Assistant OS 9.5
  Home Assistant Core:      2023.2.3

  Home Assistant URL:       http://homeassistant.local:8123
  Observer URL:             http://homeassistant.local:4357
➜  ~ 
```

Fig. 8 Terminal for Home Assistant

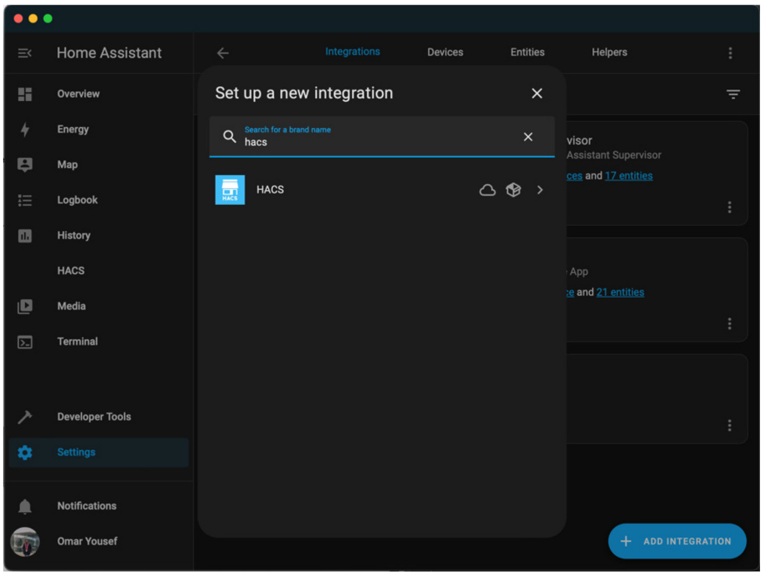
At that point, the terminal option was then available (Fig. 8).

After that, the command *cdconfig* was written to switch to the configuration directory. The next step then was to install the Home Assistant Community Store (HACS), which is an integration that finds user-made addons on GitHub and makes them available for installation. This was done by simply writing a command (also provided in the Make Magazine article) in the terminal.

wget -q -O https://install.hacs.xyz | bash

After that command was written, HACS needed to be searched for through Settings > Devices & Services > Integrations (the tab from above), and then + ADD INTEGRATION (Fig. 9).

A problem that occurred during the installation of HACS was that it could not be found while being searched for in the integrations search menu. The solution to that was simply by clearing the cache of the browser. Once this had been done, searching for HACS was no problem and it was shown in the search results (Fig. 10).



Set up a new integration

Search for a brand name: hacs

HACS

Fig. 10 Searching for HACS

To install HACS, a GitHub account was required. It would require a code to be entered to the GitHub account to activate the device. After having been redirected and logged into the GitHub account page, it was then where the HACS installation was successful and ready to use, where it had then appeared in the integrations list as well as in the left menu of Home Assistant.

MAC Address vs UUID on MacOS

The next step was to try to find the MAC (Media Access Control) addresses of the thermostat and the smart thermometer. We came to a realization that the MacOS operating system doesn't really work with MAC addresses, but it works with UUID (Universal Unique Identifier). It was basically the same idea of each device having its own unique code. The format is different than how a typical MAC address would look like. As python-eq3bt (used by Home Assistant Core) also uses bleak to communicate with Bluetooth devices and someone mentions that

"You can use the address attribute of the discovered device to create the client object. Bleak knows that MacOS uses the UUID instead of the Bluetooth address."[4]

so we might just find the UUID (with CC-RT-M-BLE) and add it to the MAC address of the configuration on MacOS.

Thermostat

UUID of Thermostat on MacOS:
7294D8FF-CCF9-9074-97E0-80BDE549787C

The first problem that occurred was that if the thermostat was connected via Bluetooth to any other device, it wouldn't show with the list of visible Bluetooth devices. Another problem that occurred was that the MacOS needed to be very close to the thermostat in order for it to detect it. The distance where it was visible was no more than 50 cm. Although we got the UUID of the thermostat with a python script, homeassistant warns us that such use needs more support and ran into some errors to get the integration work properly. Even the eq3cli tool from python-eq3bt could not work properly with UUID. A command can be used to check the availability and get some data from the thermostat.

eq3cli --mac 7294D8FF-CCF9-9074-97E0-80BDE549787C

When this problem occurred, we thought that maybe using the Windows operating system might help us, since we would be able to find the MAC address instead of the UUID, and maybe this was the problem that was stopping us from proceeding. So a switch to Windows WSL was then required to try to find the MAC address of the thermostat. Again, we found the Ubuntu (WSL) lacks Bluetooth

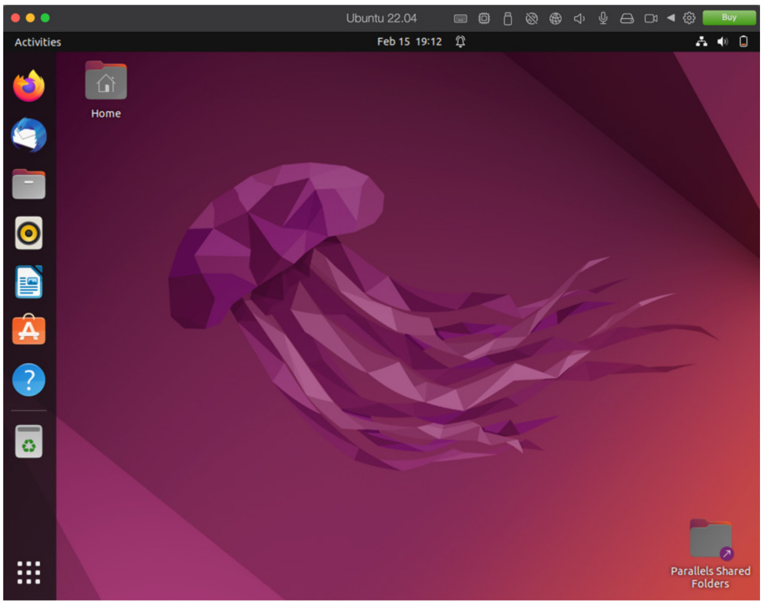


Fig. 11 Ubuntu Linux in Parallels Desktop

devices, and according to the discussion of this issue[5], we have to compile customized kernel for such use.

MAC Address of Thermostat on Windows:
00:1A:22:12:B3:A9

This was us experiencing how complex it could be without owning a Linux PC or a Raspberry Pi. Unfortunately, since the Windows WSL trial was not successful, we had to try another approach. We went back to the MacOS operating system to download Parallels Desktop with Bluetooth Sharing enabled. This would enable us to download Ubuntu Linux, which basically acted as an individual independent Linux PC (Fig. 11).

It came to our surprise that the MAC address on the Linux operating system was a different one.

MAC Address of Thermostat on Ubuntu
Linux: F9:CC:FF:D8:94:72

Although we could find the thermostat and its MAC address, the eq3cli could not connect it. It might be that the virtual Bluetooth Adapter is not stable enough for such use case.

Smart Thermometer

The smart thermometer process was the same, it also needs the adapter and the MAC address, and we faced the exact same problem that occurred with the thermostat.

3.4 Thermal Comfort & Dew Point

The optimum thermal comfort of the human body:

Temperature: 23-26 °C (Summer), 20-23.5 °C (Winter)
Relative Humidity (RH): 30-60%

This is mostly through the ANSI/ASHREA Standard 55, which is an American National Standard published by ASHREA, which establishes the ranges of the most optimum indoor environmental conditions to achieve thermal comfort.

There could be other factors affecting the thermal comfort of the human body.

- Local Climate
- Country and its Weather Conditions
- Building Type, HVAC Mode
- Body Build, Age, Gender

3.5 Room Level Automation with ESPresense

ESPresense is an ESP32 based presence detection node for use with the Home Assistant mqtt_room component for localized device presence detection. This is a

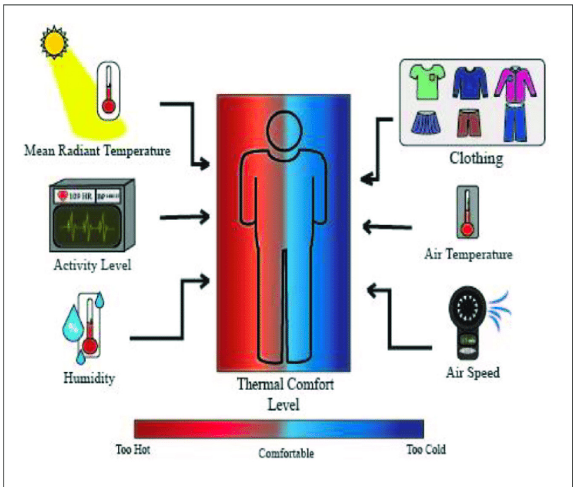
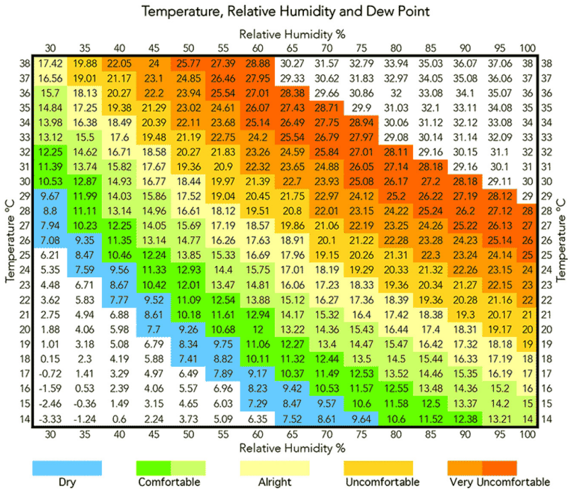


Fig. 12 Factors Affecting Thermal Comfort [6]



Temperature, Relative Humidity and Dew Point

Temperature °C \ Relative Humidity %	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100	Temperature °C
38	17.42	19.88	22.05	24	25.77	27.39	28.88	30.27	31.57	32.79	33.94	35.03	36.07	37.06	38	38
37	16.56	19.01	21.17	23.1	24.85	26.46	27.95	29.33	30.62	31.83	32.97	34.05	35.08	36.06	37	37
36	15.7	18.13	20.27	22.2	23.94	25.54	27.01	28.38	29.66	30.86	32	33.08	34.1	35.07	36	36
35	14.84	17.25	19.38	21.29	23.02	24.61	26.07	27.43	28.71	29.9	31.03	32.1	33.11	34.08	35	35
34	13.98	16.38	18.49	20.39	22.11	23.68	25.14	26.49	27.75	28.94	30.06	31.12	32.12	33.08	34	34
33	13.12	15.5	17.6	19.48	21.19	22.75	24.2	25.54	26.79	27.97	29.08	30.14	31.14	32.09	33	33
32	12.25	14.62	16.71	18.58	20.27	21.83	23.26	24.59	25.84	27.01	28.11	29.16	30.15	31.1	32	32
31	11.39	13.74	15.82	17.67	19.36	20.9	22.32	23.65	24.88	26.05	27.14	28.18	29.16	30.1	31	31
30	10.53	12.87	14.93	16.77	18.44	19.97	21.39	22.7	23.93	25.08	26.17	27.2	28.18	29.11	30	30
29	9.67	11.99	14.03	15.86	17.52	19.04	20.45	21.75	22.97	24.12	25.2	26.22	27.19	28.12	29	29
28	8.8	11.11	13.14	14.96	16.61	18.12	19.51	20.8	22.01	23.15	24.22	25.24	26.2	27.12	28	28
27	7.94	10.23	12.25	14.05	15.69	17.19	18.57	19.86	21.06	22.19	23.25	24.26	25.22	26.13	27	27
26	7.08	9.35	11.35	13.14	14.77	16.26	17.63	18.91	20.1	21.22	22.28	23.28	24.23	25.14	26	26
25	6.21	8.47	10.46	12.24	13.85	15.33	16.69	17.96	19.15	20.26	21.31	22.3	23.24	24.14	25	25
24	5.35	7.59	9.56	11.33	12.93	14.4	15.75	17.01	18.19	19.29	20.33	21.32	22.26	23.15	24	24
23	4.48	6.71	8.67	10.42	12.01	13.47	14.81	16.06	17.23	18.33	19.36	20.34	21.27	22.15	23	23
22	3.62	5.83	7.77	9.52	11.09	12.54	13.88	15.12	16.27	17.36	18.39	19.36	20.28	21.16	22	22
21	2.75	4.94	6.88	8.61	10.18	11.61	12.94	14.17	15.32	16.4	17.42	18.38	19.3	20.17	21	21
20	1.88	4.06	5.98	7.7	9.26	10.68	12	13.22	14.36	15.43	16.44	17.4	18.31	19.17	20	20
19	1.01	3.18	5.08	6.79	8.34	9.75	11.06	12.27	13.4	14.47	15.47	16.42	17.32	18.18	19	19
18	0.15	2.3	4.19	5.88	7.41	8.82	10.11	11.32	12.44	13.5	14.5	15.44	16.33	17.19	18	18
17	-0.72	1.41	3.29	4.97	6.49	7.89	9.17	10.37	11.49	12.53	13.52	14.46	15.35	16.19	17	17
16	-1.59	0.53	2.39	4.06	5.57	6.96	8.23	9.42	10.53	11.57	12.55	13.48	14.36	15.2	16	16
15	-2.46	-0.36	1.49	3.15	4.65	6.03	7.29	8.47	9.57	10.6	11.58	12.5	13.37	14.2	15	15
14	-3.33	-1.24	0.6	2.24	3.73	5.09	6.35	7.52	8.61	9.64	10.6	11.52	12.38	13.21	14	14
	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100	

Relative Humidity %

Dry · Comfortable · Alright · Uncomfortable · Very Uncomfortable

Fig. 13 Temperature, Relative Humidity

fork/rewrite of ESP32-mqtt-room.[7] ESPresense could also connect some sensors like Temperature & Humidity module like DHT22 out of the box.

Steps:

a. Install firmware to esp32 devices via usb (web-serial).
b. Configure Wifi.
c. Set a name for the station.
d. Calliberate the maximum distance and rssi expectation.
e. Setup beacon (e.g., smart phone or esp32 board) for tracking in Home Assistant.
f. Add automation & test.

Dashboard of Home Assistant

As ESPresense is still intensively under developing, it has some issues with the RISC-V esp32-c3 board. The enroll UI to get the IRK for your iOS device is missing on the M5 Stamp C3 board, while it is functioning properly with the old standard esp32 dev board. And ESPresense could not setup AP (Access Point) on esp-C3-32s-Kit with 2m, which is related to the Arduino library for esp32-c3 chips.

3.6 AR Integration

Augument Reality is a great way to interact with IoT devices intuitively.

“It is where augmented reality is the perfect solution. You can take your phone and point at any building, room or appliance to call an action. For example, I have a sauna in my garden that has to get to a certain temperature level before people get in. AR enables me to

skip multiple steps to get information. I no longer have to open the Home Assistant app and select the right building and the function I need. I can point my phone camera at the sauna and see if it is hot enough to enjoy.”[8]

3.7 Matter

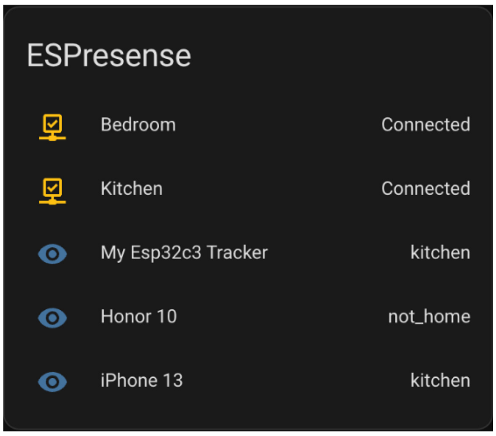
When you touch the smart home topic, you would probably find so many different lot. Now comes Matter at the end of 2022, a protocol that would integrate with almost any smart home device in the future.

“Matter is a common language for smart home devices. It’s designed to simplify everything about the smart home, from purchase to setup and everyday use. Its biggest promise is making smart devices work with each other across platforms and ecosystems, no *matter* who made them.”[9]

protocols (e.g., Wifi, Zigbee, Bluetooth 4.2/5/5.1, BLE, Lora etc.) that confuse you a

Across the Different Protocols Supported by Matter. Image: Thread Group[9]

Benefits of matter[10]:



ESPresense

Bedroom	Connected
Kitchen	Connected
My Esp32c3 Tracker	kitchen
Honor 10	not_home
iPhone 13	kitchen

Fig. 14 Dashboard for ESPresense

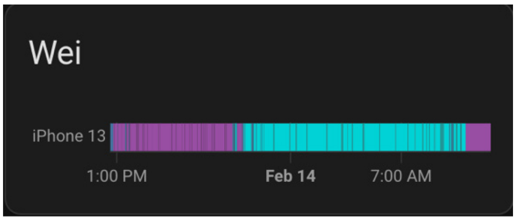
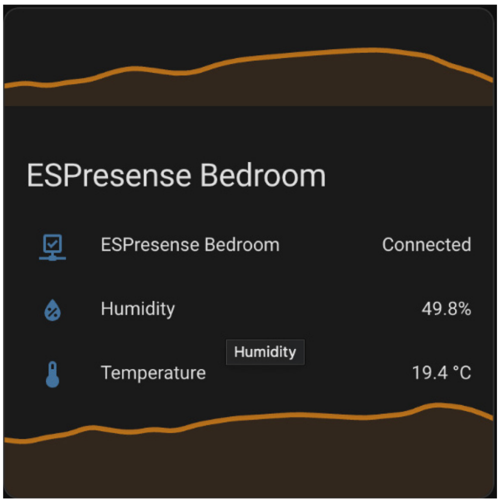


Fig. 15 History for iBeacon device (iPhone)



ESPresense Bedroom

ESPresense Bedroom	Connected
Humidity	49.8%
Temperature	19.4 °C

Fig. 16 Integration with Temperature and Humidity sensor (DHT22)

a. local connection – Internet is optional
b. the radio protocol thread – mesh network
c. easy setup – QR code
d. security and privacy – encryption, device certificates and blockchain technology

Home Assistant supports Matter with a beta addon Matter-server. For esp32 chips, Espressif has esp-matter sdk[11], so that you could turn an esp32 dev board into a matter device.

4. Discussion

4.1 Complexity

During our experiment, we came across many issues that is not stated in the guidelines or official documents. Some issue need expert knowledge and time to resolve. We also realized that the interdisciplinary knowledge sharing is quite difficult. The terms and methods for IT-expert & non-IT-expert are so different, which requires much effort and time to understand and have enough confidence to handle by general people. When someone asks for help in the forum or posts an issue on GitHub, contributors and other people need more special details (OS, version, steps, errors, log, screenshots, etc.) to understand the problem, and they also need to describe it in an easy way to help tackle the problem.

Compared to Tuya TRV603 Wifi Thermostat[1], our solution seems more complex to setup, less stable and costs more (without include the cost of a Raspberry Pi). Through this unsuccessful experiment of smart home, we do learn a lot and have a better understanding about the IoT devices and it prepares better us for the digitalization of traditional construction industry.

4.2 Customization

Home Assistant has a community store for add-ons to add more features to integrate into smart home automation.

4.3 Multi-Platform Support

As matter comes as an interoperate protocol, new products could easily support multi-platform, which is great for customers.

4.4 Cost & Saving Potentials

According to nest energy saving white paper[12], smart thermostats could save 10-15% of cost for heating. While the smart devices are quite expensive, it may not be so attractive for a small single apartment to invest such automation if the rent is short. In the long run, however, they could save a very considerable amount of money and energy.

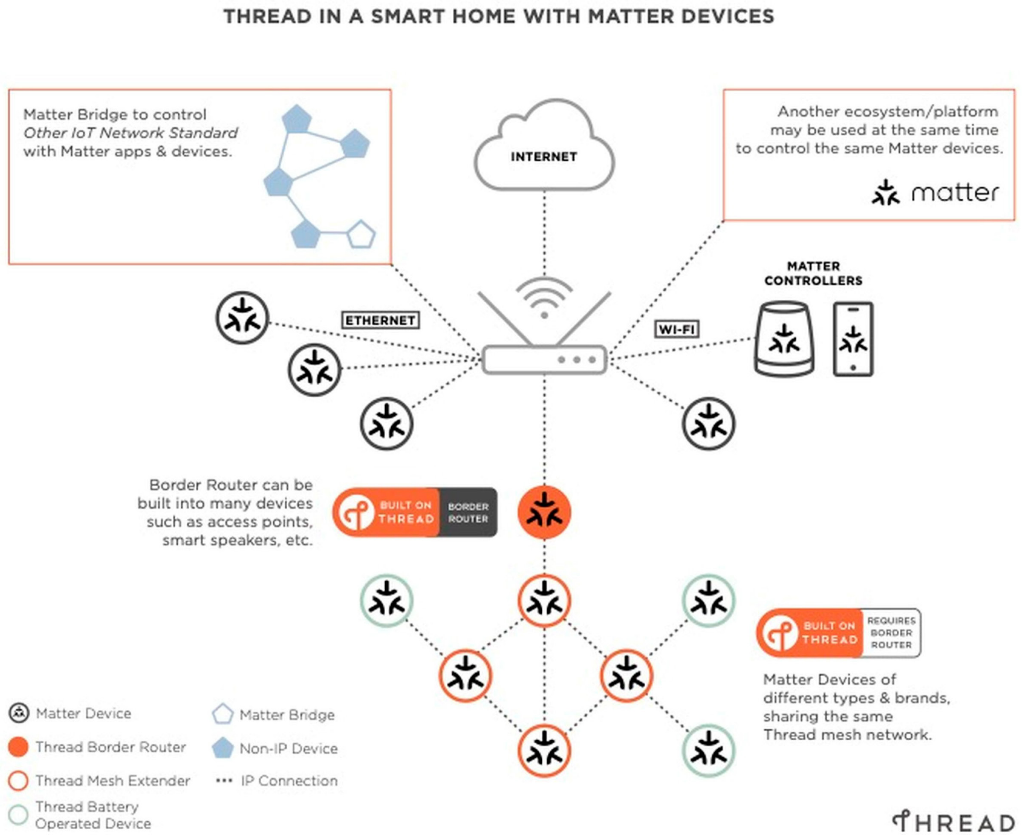

THREAD IN A SMART HOME WITH MATTER DEVICES
Matter Bridge to control Other IoT Network Standard with Matter apps & devices.
INTERNET
Another ecosystem/platform may be used at the same time to control the same Matter devices.
matter
MATTER CONTROLLERS
ETHERNET
WI-FI
Border Router can be built into many devices such as access points, smart speakers, etc.
BUILT ON THREAD
BORDER ROUTER
BUILT ON THREAD
REQUIRES BORDER ROUTER
Matter Devices of different types & brands, sharing the same Thread mesh network.
Matter Device
Thread Border Router
Thread Mesh Extender
Thread Battery Operated Device
Matter Bridge
Non-IP Device
IP Connection
THREAD

Fig. 17 How Devices Will Communicate Across the Different Protocols Supported by Matter. Image: Thread Group[9]

4.5 Security

The prospect of automation could be interesting and enjoyable, provided that smart devices are protected from potential malicious users.

Smart Home Security Tips for Customers [13] :

- Use the screen lock on your smartphone.
- Ensure all your computers and smartphones are password protected (consider using a password manager, e.g., Bitwarden).
- Ensure your main computer account is not at an administrator or root level.
- Change the default username and password on your router.
- Use firewalls on any computers and on your router (consider setting up a guest network for smart home devices).
- If your existing router doesn't offer you good security features, replace it.
- Use a strong security software on your computers and smartphone.
- Always run security patches and updates and keep your software up to date.

For smart home developers, rewriting/implementing a software with rust language could also reduce risks of exploits and many security issues. TockOS is a promising alternative for Free-RTOS. A good start could be the Book *Getting Started with Secure Embedded Systems: Developing IoT Systems for micro:bit and Raspberry Pi Pico Using Rust and Tock*[14].

4.6 Inspirations for Construction

Although customers could setup smart home to an existing building, there could be a lot of benefits to consider fundmental smart hubs and sensors in the stage of design and install during the construcion stage. For example:

- Extra plugs/power supply, for micro stations/wifi mesh
- Smart plugs, which could control and collect eletricity comsuption, for non-smart devices
- Smart bulbs, with an IR sensor, which could be used for room level presence
- Extra space for smart curtain motor
- Smart sensors to provide maintenance information
- BIM data for AR/ Digital Twins applications

5. Conclusion

5.1 Challenge & Trend

Setting up smart home system is still a complex task even for experts. With matter coming to production, smart home automation could be easier and easier for new customers and hobbyists. We learned a lot through spending a lot of effort experimenting and researching for so many hours, whether individually or together. Even though this was unfortunately an unsuccessful experiment, we believe that it would be a benefit for us in some sort. We are pretty sure that it would give us a head start when the automation and application of smart homes would become easier and more popular in the very near future.

5.2 Community & Open-Source

As smart home is so complex and not mature, community support is a good way to get help from experienced experts and other customers. Open-source community is great for customization to get the devices smart for your need.

5.3 AR & AI

AR application for smart home might become a must feature for future smart home, espcially when AR glasses come to daily life. As smart home collects so much data about your daily acitivities, AI companies would have great interests to help you find a better solution to build more automation for your smart home.

5.4 Construction

If smart home fundamentals could be considered during the construction processing, the residents could enjoy a better life.

6. References

[1] “Tuya TRV Wifi Thermostat,” AliExpress. https://www.aliexpress.com/item/1005004755554233.html (accessed Feb. 08, 2023).

[2] Behling, Heinz, “Heizung unter Kontrolle | Make Magazin | Heise Magazine,” vol. 1/2021, p. 100.

[3] H. Assistant, “Install Home Assistant on a MacOS,” Home Assistant. https://www.home-assistant.io/installation/macos/ (accessed Feb. 11, 2023).

[4] “Finding the MAC address on macOS · Issue #284 · hbldh/bleak,” GitHub. https://github.com/hbldh/bleak/issues/284 (accessed Feb. 09, 2023).

[5] “Bluetooth not supported (Built-in and USB-BT Adapter) · Issue #242 · microsoft/WSL.” https://github.com/microsoft/WSL/issues/242 (accessed Feb. 15, 2023).

[6] A. Ghahramani, P. Galicia, D. Lehrer, Z. Varghese, Z. Wang, and Y. Pandit, “Artificial Intelligence for Efficient Thermal Comfort Systems: Requirements, Current Applications and Future Directions,” Front. Built Environ., vol. 6, p. 49, Apr. 2020, doi: 10.3389/fbuil.2020.00049.

[7] “ESPresense,” ESPresense. https://espresense.com/ (accessed Feb. 06, 2023).

[8] “Expert interview: home automation using augmented reality,” Overly, May 23, 2022. https://overlyapp.com/blog/expert-interview-home-automation-using-augmented-reality/ (accessed Feb. 06, 2023).

[9] J. P. Tuohy, “What Matters about Matter, the new smart home standard,” The Verge, Dec. 14, 2021. https://www.theverge.com/22832127/ma

tter-smart-home-products-thread-wifi-explainer (accessed Feb. 09, 2023).

[10] "Benefits of matter at a glance," matter-smarthome, Feb. 13, 2023. https://matter-smarthome.de/en/benefits/ (accessed Feb. 14, 2023).

[11] "Espressif's SDK for Matter." Espressif Systems, Feb. 09, 2023. Accessed: Feb. 09, 2023. [Online]. Available: https://github.com/espressif/esp-matter

[12] "Save energy and live comfortably with Nest thermostat savings," Google Store. https://store.google.com/ideas/articles/nest-thermostat-savings/ (accessed Feb. 06, 2023).

[13] "How safe are smart homes?," www.kaspersky.com, Jan. 13, 2021. https://www.kaspersky.com/resource-center/threats/how-safe-is-your-smart-home (accessed Feb. 08, 2023).

[14] A. Radovici and I. Culic, Getting Started with Secure Embedded Systems: Developing IoT Systems for micro:bit and Raspberry Pi Pico Using Rust and Tock. Berkeley, CA: Apress, 2022. doi: 10.1007/978-1-4842-7789-8.

construction
robotics
www.cr.rwth-aachen.de

international
M.Sc.programme