Comparing smart home middleware concepts and framework solutions for context aware energy management capabilities

Jana Naderhoff
Furtwangen University
Faculty of Computer Science, MOS2
Furtwangen, Germany
jana.naderhoff@hs-furtwangen.de

Furqan Yousaf
Furtwangen University
Faculty of Computer Science, MOS2
Furtwangen, Germany
furqan.yousaf@hs-furtwangen.de

Musa Aliev
Furtwangen University
Faculty of Computer Science, MOS2
Furtwangen, Germany
musa.aliev@hs-furtwangen.de

***Abstract*—Global warming, climate change and rising energy costs are the problems we face today and which we need to work over. In 21th century with highly developed knowledge in IT technology and all the smart devices we use daily it is obvious that energy consumption is rising enormously. The level of consumption often exceeds our needs which results in overproduction of energy and excessive emission of CO2. Smart devices and technologies must learn to help us save energy by reducing and optimizing its consumption where it is possible. In this paper, we examine the use of smart home middleware capable of context awareness in energy management solutions. We examine proposed concepts in literature and attempt a comparison to current available smart home middleware frameworks, looking into the extent of context awareness they are capable of and how solutions proposed in literature could be reproduced for energy management application. Furthermore, we attempt to evaluate whether the current extent of context awareness needs to develop further for this purpose.**

Keywords—smart home, energy management, framework, context awareness, middleware, OpenHAB, Home Assistant, wireless sensor network

I. INTRODUCTION

The climate change and all its mostly negative side-effects are becoming reality, and it is obvious that mankind, or rather the relentless activity of mankind, is the cause. Global warming and resulting climate change are going to affect not only the environment but human household budgets too [1]. The interesting fact is that we all cause these problems, directly or indirectly. Even our homes are sharing this responsibility with heavy industry and mobility caused pollution. Ordinary households for example contribute to global warming by consuming energy for heating or lighting. The rising usage of multimedia devices or ordinary household electrical devices are the next factor in global greenhouse gas production. According to statistics delivered by Statista (see Fig. 1 below), heating and power generation account for up to 42% of all CO2 emissions worldwide [2].

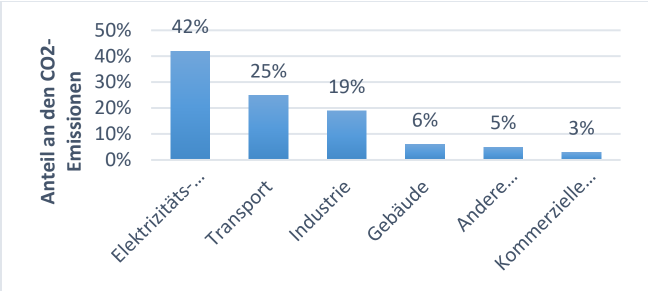

Figure 1: CO2 emissions worldwide in %. Statista

Though, this varies from country to country. According to the same Statista source the energy-related CO2 emissions accounted for the vast majority of all German CO2 emissions in 2020, at around 93 percent [3]. CO2 emissions are greenhouse gases and air pollutants that are produced during conversion of energy sources (e.g., into electricity and heat). The amount of CO2 emissions worldwide caused by energy and heating in 2018 was about 14 billion tons [4].

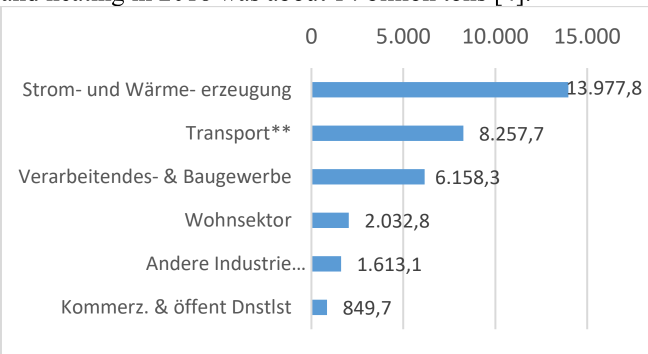

Figure 2: CO2 emissions worldwide in tons. Statista

Tt is obvious, that every contribution in a reduction of CO2 emissions counts. A research done by Morgan Stanley Research reveals “that the efforts we make at home may be just as important. Daily household tasks, from using the washer to turning on heating, account for 17% of the world's carbon-dioxide emissions” [5].

II. INNOVATING HOUSHOLDING IS THE FUTURE

Innovative householding, like Smart Home, could potentially help reduce carbon gas emissions produced from households worldwide. At least ubiquitous computing and Internet of

Things make possible through the ability of different smart devices to communicate with each other or with other interfaces in framework. The possibilities in development of innovating householding are growing due to technical innovations. In another report done by Morgan Stanley Research and published under "Consumers and Climate Change: The Home Energy Efficiency Opportunity" title, the researchers support these ideas. The authors of the report focus on fast developing Internet of Things (IoT) as great chance to make our houses smarter and more energy efficient, “passing cost savings to consumers, creating opportunities for investors and perhaps most importantly, benefiting the planet in the process, according to the report” [6].

III. Smart Home

A smart home is a concept represented by a dwelling equipped with communication networks, sensors, appliances, and devices that can be remotely monitored, accessed, or controlled and which provides services that respond to specific user needs [7]. As said, this sensor-based technology gathers necessary information inside and outside of the household and processes that information to centralized or decentralized decision-center which presents it in user interface where a user can take appropriate decisions. The problem is – the turn to smart home living till now is done mostly not to save the energy consumption but for convenience of life. This point of view is supported by researchers in different research papers such as “Learning to live in a smart home” [8] or “Culture, energy and climate sustainability, and smart home technologies: A mixed methods comparison of four countries” [9]. According to authors the primary goal of using smart home technology is leisure and convenience and can even “increase energy consumption" [8]. The authors refer to the facts that sensor infrastructure needs to be in standby mode 24 hours per day and this requires energy supply for this time. As a result, the consumption raises there where it was expected to be sinking. An ability of a system to adapt itself to changing environment by collecting data about context, in which this data is generated, analyze this data, and adjust smart device behavior to the changing context, has promising future.

IV. Context awareness

The ability of smart devices and smart applications to identify current operating conditions or context and adapt their behavior on the basis of the context is termed as context awareness [10]. Under the context we simply understand the description of the current situation as bound by the environment, while context awareness utilizes the context to adapt and change the services present in the environment. The environment becomes smart and delivers better services to the end-user [11]. Context-awareness on the other hand is an important characteristic of ubiquitous computing [12]. In a smart home framework context plays data source role. Today a huge number of smart home devices generate raw data about their work environment, this raw data needs to be analyzed and interpreted to have meaningful information from it such as location or time (this information called contextual information) to be feasible [13]. The raw data that sensors collect, such as preferred inhouse temperature, time spent in front of TV, listening to music, cooking, bathing or even sleeping times is enormous. All this data needs to be gathered, aggregated, distributed. And smart home framework is playing central role here. Current smart home frameworks can already work with this data. The problem is that context changes within short periods of time and a normal human is not always able to react to these changes. So, the great challenge for the context-aware systems within smart home domain is an ability to understand the changing context and react to changes according to pre-configured rules or even on their own. This would however need involvement of machine learning on the framework level so that the middleware could interpret the contextual information. Usually, the framework itself is a complex infrastructure which consists of different hardware and software components communicating with each other via network of different art and standards. For the context-aware systems is the middleware especially interesting.

V. Context middleware

Context-Aware Middleware is defined as software system that provides an abstracted layer among the context-aware applications and the operating systems [14]. The context-aware architecture has different architectural styles, it can be distributed, centralized, or layered [13]. In a distributed architecture the components of architecture take over the role of data aggregator, analyzer and actuator. The action is done not centrally but locally in this case. In centralized architecture the context information is delivered to central computing unit which analyzes the data, assigns it to exact devices and takes necessary changes if needed. Layered architecture is sophisticated through usage of different layers like in our next example. There are number of ideas and concepts how the context-aware middleware for smart home should look like but very few which have a reference to energy saving contextual environment. It is expected that context-aware middleware would enable a way to collect data and adjust itself to new circumstances by learning from this data. The available studies propose different art of context-aware middleware. Hoque et al. in “Middleware aided context-aware service for smart home“ [15] propose a middleware layer which is shown in Fig. 3. In this middleware design the contextual data gets through different layers till it gets registered.

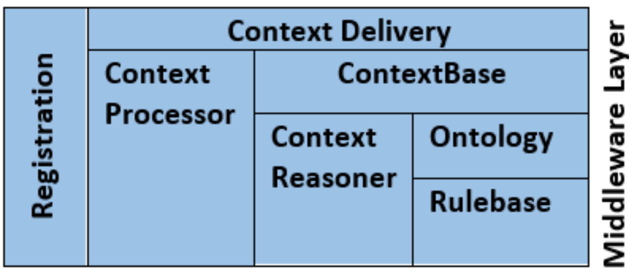

Figure 3: Layered middleware for smart home. “Middleware aided context-aware service for smart home“.

Context processor has the role of high-level context-builder, and it uses several context combinations for that. Context reasoner shall also make high-level context but uses rules and domain ontology for this purpose. According to authors’ idea the context processor with context reasoner has a role context provider. Registration service provides facilities to declare capabilities of context providers. Registration service would then store name, type of context and access point of context source in context base. The result of this type of middleware work should be smart home services which would react to context by using context with IF-THEN condition. The conceptual middleware delivers the aggregated and analyzed information to home server which decides what action to take as next. The use of IF-THEN rules gives a grade of automation to such middleware, but it is not yet fully automated. The grade of automation here is limited by the programmer who created the rules.

Another view on context-aware middleware give us Mahmud et al in their research article "Power Profiling of Context Aware Systems: A Contemporary Analysis and Framework for Power Conservation" [16]. A centralized art of architecture in this concept consists of contextual data providers, context-aware middleware system and services generated by this system like shown in Fig. 4.

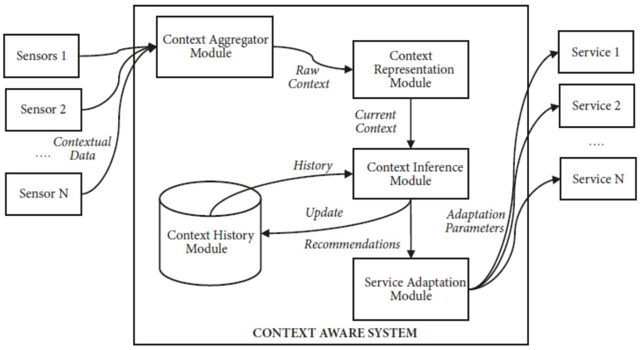

Figure 4: Centralized context aware framework. "Power Profiling of Context Aware Systems: A Contemporary Analysis and Framework for Power Conservation".

This type of middleware is more advanced than the one in layered example. The context information in proposed concept gets aggregated from all sensor devices as raw context and then representation module presents it to inference module which compares it with existing context information. If something in context changed the change will be merged in history module database. Inference module is AI based component which performs data-change analysis and makes recommendation or triggers a rule. This concept was created for smartphone battery saving but can be also used in smart home frameworks when implemented to recognize the context changes and react to these changes. For example, through summoning information about current outside/inside temperature, daylight times and weather forecast plus presence of inhabitants in the household such context-aware system can adjust the energy consume even without interference from the users. The problem is – these frameworks are still in concept form and have not been yet physically implemented. Sure, there are many smart home frameworks which have been developed and are even available for try and use. The question is how good the usability/readiness of these frameworks and especially the middleware part of these is. Can a random user implement it in his/her house? What is the level of automation these frameworks offer? In our research we are going to have a closer look on two of them as next.

VI. RELATED WORKS

Han and Lim [17] propose a „Smart Home Energy Management System (SHEMS) based on IEEE802.15.4 and ZigBee" [17, S. 1417] which employs their own routing protocol „DMPR (Disjoint Multi Path based Routing) to improve the performance" [17, S. 1417]. They recognize context awareness as important in order to minimize necessary user interaction with a ubiquitous computing environment and identify the context aggregator system component as necessary in order to „integrate diversified sensing information to perceive the current situation in surrounding environment" [17, S. 1417]. They propose to employ a wireless sensor network to gather environmental information to feed to the context aggregator, which processes the incoming data and evaluates it, deciding what actions should be executed by connected actuators, making use of electronic switches and infrared signals [17]. The context aggregator component requires every router device in the network to have an entry in a context dictionary component corresponding to the context aggregator [17]. As this would normally result in broadcasting overload with multiple sensors trying to discover the context aggregator simultaneously, Han and Lim employ ZigBee to have the context aggregator announce its presence instead, setting up entries in a component called context broker which functions as a relay [17]. Their context aggregator component is supposed to handle both „implicit and explicit input" [17, S. 1421] from many different sensors, setting up a record which can be polled and subscribed to by other components [17]. The aggregated information is to be processed by what Han and Lim call the „Information Extractor" [17, S. 1421], consisting of the three components feature extractor (responsible for abstracting or interpreting context data, for example turning a set of coordinates into a street name), data mining (responsible for helping „to build contextual databases" [17, S. 1421] that facilitate knowledge sharing between agents, updated by agents whenever they acquire new knowledge) and context predictors (responsible for first selecting the type of context a piece of information from the shared knowledge base is supposed to infer, such as the location of a person) [17]. Next, Han and Lim propose a „Service Extract Engine" [17, S. 1421] to further process information in cases where the possible context is ambiguous, utilizing Jess Reasoning Rules to try and infer context with any gathered supporting facts, creating an RDF (Resource Description Framework) representation [17]. Lastly, by applying system behaviour rules to the determined context information, Han and Lim propose smart home services to make use of it. One of their given examples is a climate control system activating or deactivating devices such as air conditioners depending on room temperature, occupant location and whether a window has been opened or closed [17]. Another example given is room illumination control by activating, dimming/brightening or deactivating lights depending on sunlight levels and occupant location [17]. Han and Lim's idea of context-aware energy management revolves around using context in order to provide just the necessary amount of energy-consuming resources in the exact location where they are determined as being needed, reducing excess.

Naji et al. [18] propose their own implementation of a „Wireless Sensor Network-based Energy Management System“ [18, S. 1], making use of „the Context-Based Reasoning (CBR) model“ [18, S. 1] and „a Finite State Machine (FSM) leveraging on context-related events“ [18, S. 1] which they claim reduces overall system strain. Furthermore, they implement their own „Energy Aware Context Recognition Algorithm (EACRA)“ [18, S. 1] to have sensors only send data when needed to avoid redundancy and thereby save sensor battery power [18]. The implementation they propose consists of an appliance monitoring framework, a „database to archive user behavior and environmental data“ [18, S. 2], and appliance control software [18]. Instead of employing one energy policy for an entire building, Naji et al. seek to adjust it per room based on each room's context, using CBR to model the different rooms [18]. They, like Han and Lim, use custom sensors and actuators to acquire context information which „forward data to the server via an intermediate gateway device that connects the WSN to the control server“ [18, S. 4]. Naji et al. divide their energy management system into layers, starting with a „physical layer“ [18, S. 5] for interaction (activating, deactivating, regulating etc.) with physical objects, „composed of WSN and IoT, such as sensors and actuators“ [18, S. 5], followed by a „software layer“ [18, S. 6] which is responsible for contextual decisions based on sensor data and energy policy [18]. Several communication protocols are employed between layers: „WiFi, Zigbee, Z-wave, Ethernet, Bluetooth, and Infrared“ [18, S. 6]. To determine context, Naji et. al combine sensor data with „static parameters that describe the building type, room type, season, current day, and part of the day“ [18, S. 13] which are received from a database [18]. Combined context identification results are then being fed into „Finite State Machines“ [18, S. 13] which are responsible for „appropriate control decision“ [18, S. 13], their algorithm „guided by the energy policy set by the building manager“ [18, S. 14]. Three types of knowledge are being named which converge into context: „action knowledge“ [18, S. 14] which is „operational intelligence for a specific situation […] coded with logic rules or learned using machine learning“ [18, S. 14]; „transitional knowledge“ [18, S. 14] which „states when a transition to another context is needed“ [18, S. 14] and is rule- or trigger-based; and „declarative knowledge“ [18, S. 14] which „describes some aspects of the context to include some of the pre-acquired experience for the context – for example, the number of inhabitants, room size, or room usage schedule.“ [18, S. 14] Their three types of knowledge help create context groups for the CBR model needed to infer context [18]. By drawing from the groups and combining with sensor data, room context is generated and evaluated in junction with the room's energy policy (the rules of which are represented by FSM states), resulting in a decision on how to affect an appliance state [18]. Sensor energy consumption itself gets reduced by having the EACRA algorithm check whether declared knowledge and contexts in the CBR model changed, only sending new information if it changes [18].

Further similar proposed solutions for context-aware energy management include works from Chen, Liu et al. [19], Lillis, O'Sullivan et al. [20], Lu, Wu et al. [21], Nam, Rho et al. [22], Weng, Wu et al. [23] and Raja Vara Prasad, Rajalakshmi [24].

VII. INTERMEDIATE CONCLUSION

Having looked into available literature, we conclude that there is not yet a way of implementing a context-aware energy management middleware system that is well accessible to non-researcher users. Proposed solutions involve experimental custom-made components, algorithms and protocols which we think tend to not be realistically replicable by most people, even if they are willing to go to greater efforts to set up their own smart home environment, due to steep technical knowledge requirements.

What we consider a more accessible approach is to choose from current available open-source smart home middleware solutions which can be customized or built upon for multiple purposes, including context awareness, and looking into their context awareness capabilities, examining if they already include components similar to those proposed in literature so that it might be feasible implementing context-aware energy management solutions with them.

Furthermore, we notice that context awareness as implemented in the presented example solutions is mostly if not entirely based on preset rules that enable automation, with the resulting systems not yet able to learn from user behavior for more extensive predictive actions. Considering the fast advances in machine learning technology, we also examine available open-source smart home middleware solutions for the extent, if any, of their machine learning and predictive action capabilities.

VIII. SMART HOME FRAMEWORKS

As mentioned above smart home frameworks have mainly three types of architecture – distributed, layered and centralized. Smart home automation system architecture influences the characteristics and capabilities of the system itself. There is a number of frameworks available on the market and they act in common very similar. IoBrocker, Domoticz, openHAB, Home Assist and many others. Examining all of the is not our purpose, moreover the point of interest lays on middleware as the main agent for contextual information analysis. That's why we concentrate on middleware available in openHAB and Home Assistant. On the premise of these, we examine concepts and components needed to process the smart home automation work.

Smarthome middleware relies on physical device, database, event bus and rule engine. To present the information it needs also UI.

- **Physical device**: for example, a temperature sensor, should be addressed in the home automation system. Home Assistant just alludes to a physical device as 'device', openHAB utilizes the name 'thing'. however overall physical devices are represented as the 'object' data type.
- **Database:** it is the main part to store historical data of devices. It is also used to save the current state of the devices and other things within system. Database is also used to track all devices. Both Home Assistant and OpenHAB supports numerous database systems.

- **Event bus**: it is found in both openHAB and Home Assistant, event bus plays an important role to make the asynchronous communication between the components possible. It is also used to listen to events generated by devices or by the system itself.

- **Rule engine:** It plays an Important role for the context awareness, as it receives context information through different sensors or devices and it enables the automation of the home to define and execute rules. OpenHAB and Home Assistant provide their own components to wire together devices and events.

- **Web-based user interface**: It is not only one option to interact with smart home system but the most popular option to interact with. While Home Assistant leans heavily on a single user interface which is highly personalizable, openHAB supports multiple, entirely distinct, user interfaces each with their own strengths and weaknesses.

A. OpenHAB framework

OpenHAB is an automation system for homes based on Java allowing the users to control multiple devices and services remotely having different communication protocols. This is an open-source system that can be operated on any platform that is capable of operating Java Virtual Machine (JVM) for example mobile devices and personal computers [25].
OpenHAB is based on an IOT framework called Eclipse Smart Home (ESH). ESH is flexible, modularized, Java-based smart home framework through which assisted living conditions in heterogeneous environment are accomplished. Eclipse Smart Home is not limited on defining a single communication protocol. But it supports multiple communication Protocols that makes the framework to be compatible with growing smart home market [26]. OpenHAB is a vendor independent framework [27]. Figure 5 depicts a high-level overview of the OpenHAB.

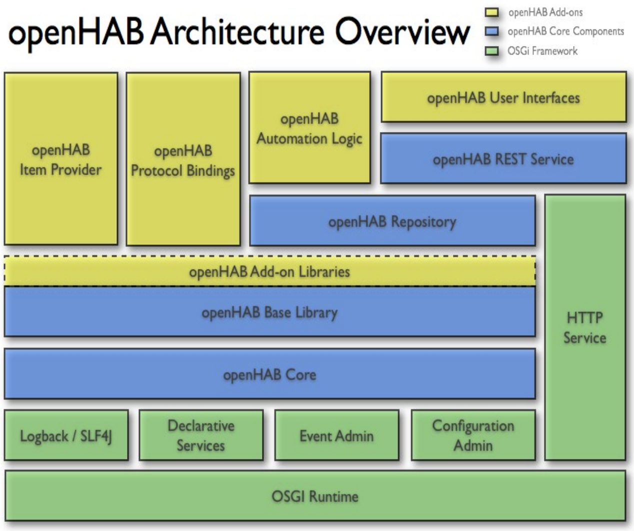

Figure 5: High-level overview of OpenHAB architecture

OpenHAB Framework Components:
OpenHAB consists of different components such as Things, Channels, Items, Bindings and Links and items. The services and physical devices which are connected within the system are known as the things. These things are used to provide the functionalities of the devices and services via one or multiple channels. The Physical layer consists of two building blocks things and channels. Items can be demonstrated as the abstract translations of the functionalities given by the channels. The automation and interaction processes are operated within the item level [28]. The most important part is middleware layer where all the automation rules are implemented.

Automation rules in OpenHAB
For managing automation rules, OpenHAB offers several built-in tools like Xtend, a flexible and expressive dialect of Java that compiles into readable Java 8 source code. OpenHAB provides an easy way to create automation rules. Instead of building its own manager like Home Assistant did, OpenHAB uses Blockly. It is an open-source tool maintained by Google that uses JavaScript to create visual block programming languages and editors.
Although the implementation of Node-RED is a bit less smooth than in Home Assistant, it still works well since all devices and states are exposed ia MQTT. In addition, pure JavaScript can also be used to write automation rules, but this is not at all intuitive [29].

Rule Defining
During the setup of smart home system, rules need to be defined to control the environment of smart home system.
Rules are defined in the rule file. One rule file may contain many rules and all rules of this file can access and exchange variables with each other. OpenHAB has different rule files for different use-cases and they can be defined in two different ways: user interface (UI) based and integrated development environment (IDE) based. OpenHAB offers support of VS Code extension to define rules using IDE.
Rule Syntax looks as in Figure 6:

```
rule "<RULE_NAME>"
when
    <TRIGGER_CONDITION> [or <TRIGGER_CONDITION2> [or ...]]
then
    <SCRIPT_BLOCK>
end
```

Figure 6: Automation rule syntax in OpenHAB

Rule Triggers:
Before applying a rule to perform some action it needs to be triggered. There are different types of triggers in OpenHAB [30].
Item(-Event)-based triggers: They react on events on the openHAB event bus, i.e. commands and status updates for items [30].
Member of(-Event)-based triggers: They react on events on the openHAB event bus for Items that are a member of the supplied Group [30].
Time-based triggers: They react at special times, e.g. at midnight, every hour, etc [30].

System-based triggers: They react on certain system statuses.
Thing-based triggers: They react on thing status, i.e. change from ONLINE to OFFLINE [30].

These different categories of rule triggers listen for some specific event, on status changes or status updates and then trigger the rule according to received command. After the rule is triggered, the state of specific item is manipulated, for example switching lights on or off under some conditions.

B. Home Assistant

Home Assistant is an open-source smart-home framework based on Python 3 as backend and Polymer as the frontend component [31]. It was invented by Paulus Schoutsen. This platform is used in the home to track and control connected gadgets. Different types of sensors, devices, and home automation lines can all be successfully integrated with Home Assistant. There are presently over 1500 distinct integrations available. New devices, adapter protocols, user interface alterations or extensions, and the integration of external services are all supported through these integrations. Home Assistant is mostly configured through the usage of YAML-configuration files. Instead, the user interface is being enhanced with new configurable choices. Figure 7 depicts a high-level overview of the Home Assistant architecture. The figure 7 from Home Assistant source [32] shows the architecture:

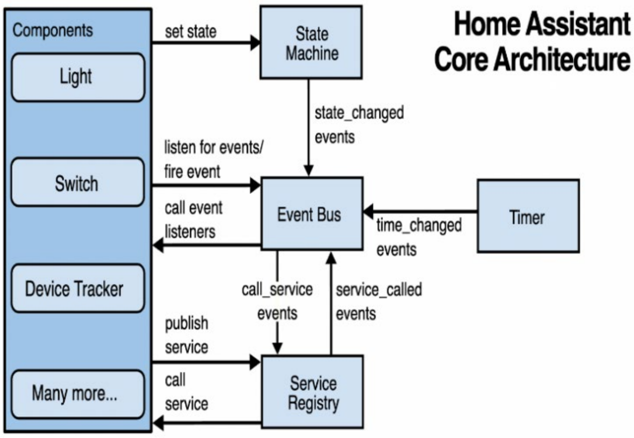

Figure 7: High-level overview of Home Assistant

Home Assistant Components:

The Event Bus, State Machine, Service Registry, and Timer are the four essential components of HA. The Event Bus, which is utilized to enable the firing and listening of events, is the heart of the HA. Once a state has been modified, the State Machine can monitor it and send a state changed event to the Event Bus.

Automation rules in Home Assistant

Home Assistant offers various automation rules but the simple one is using YAML, a data serialization language. There two approaches: first are Python-style indentations which indicate nesting and the second one uses a more compact format where [] used for lists, {} is used for maps this way making YAML 1.2 a superset of JSON. According to the article written by A. Brice, "indentation is needed to specify relationships using YAML and YAMLLint or built-in configuration checker should be used when changing the configurations" [31].

Node-RED – one more way to manage the rule automation in Home Assistant. Node-Red is a flow-based development tool for visual programming. It was originally developed by IBM to help wiring together hardware devices, APIs and online services as part of the Internet of Things. It won popularity due to its exceptionally visuality, easiness by change and deploy operations.

I. **App-daemon:** This tool stands over all other automation tools in Home Assistant due to its powerful functionality. In his web article "Best of open source smart home: Home Assistant vs OpenHAB", A. Brice says about this tool: "App-daemon brings the power and flexibility of Python to manage automation rules. This yields many benefits on things that can program but will require a good knowledge of Python"[31].

Defining automation:

Automation rules also need to be defined in Home Assistant like in OpenHAB. But in home assistant these automation rules are defined and executed differently. Automation rules consist of trigger, action and optionally combined with a condition. The first part is trigger, that defines the events to trigger the automation rule. For example, if a person arrives home and that event is observed in Home Assistant through observation of state of person changed form 'not_home' to 'home'. Second part of automation rule is condition, that limit an automation rule to work in specific use case. A condition checks against the current state of system. For example by checking system time to realize weather sun is set or not. The third part of automation rule is action, which comes into use when a rule is triggered and conditions are satisfied. For example, it turns the light on if sun is set [33].

Home Assistant provides an UI based automation editor to define automation rules to operate the smart environment autonomously. The step-to-step guide is available online in the documentation of Home Assistant.

Comparison results:

The above discussed both frameworks considered to be the best frameworks for smart home automation among many others. As we could observe, both OpenHAB and Home Assistant frameworks differ from each other in many aspects but the ultimate objective of both is same - to facilitate the users which provide the integration of different devices and different services together to automate different "things" according to the needs of user. Though home automation through frameworks like openHAB or Home Asssitant is primarily designed for inhabitant convenience and leisure purposes through different approaches it still has great potential to improve also mechanisms of energy management, and especially in energy saving. The disatvantage of these smart home frameworks is low userfriendliness. Though, Home Assistant seems to be on the right path in this question. In comparison to the configuration language used in OpenHAB, which is based more on Java programming language, which requires the user own basic knowledge about concepts of Java, Home Assistant's configuration language is much easier to understand, learn, and implement through understandable graphical user interface, which make it possible to apply the automation rules even if the user has very few or no knowledge in programming languages. The mobility support is present in both frameworks. Both allow the users to control smart home framework devices remotely through cloud services. The only difference is that OpenHAB service is free of charge where Home Assistant cloud service requires subscription. The current frameworks available on the market mainly support only performing actions according to the rules predefined by user and take only limited context information to trigger these rules.

II. Conclusion and Future Outlook

After having examined existing proposed energy management solutions and having compared the context awareness capabilities of current available middleware frameworks, we conclude that they offer solid rule-based contextual decision-making. However, this should be developed further to embrace similarly accessible implementation for machine learning-based context awareness approaches. By being capable of learning from users' long-term energy consumption behaviour, we speculate that more accurate predictions about future energy consumption needs can be made. This could potentially allow an integration of such home energy management systems with future infrastructure such as smart grids. By becoming aware of a close approximation of how much energy is needed where, when and by who, information could be relied to smart grid technology to help optimize production and distribution which in turn helps with resource conservation. In our view, it is such learned adaptability that will need to be researched in order to prepare context-aware smart home energy management systems for future technological developments – and in order to combat climat change caused by excessive energy production emissions, combining better optimization on an individual household basis with an overall rise in renewable energy sources.

References

[1] M. Nassereddine et al., „Electrical energy management for advance smart home systems: Introduction“ in 2016 3rd International Conference on Renewable Energies for Developing Countries (REDEC), Zouk Mosbeh, Lebanon, 7/13/2016 - 7/15/2016, S. 1–6, doi: 10.1109/REDEC.2016.7577506.

[2] Statista, CO2-Ausstoß weltweit nach Sektoren | Statista. [Online]. Verfügbar unter: https://de.statista.com/statistik/daten/studie/167957/umfrage/verteilung-der-co-emissionen-weltweit-nach-bereich/ (Zugriff am: 17. Februar 2022).

[3] CO2-Emissionen in Deutschland | Statista. [Online]. Verfügbar unter: https://de.statista.com/statistik/studie/id/6920/dokument/klimawandel-co2-statista-dossier/ (Zugriff am: 17. Februar 2022.933Z).

[4] Statista, Weltweite CO2-Emissionen nach Sektor 2018 | Statista. [Online]. Verfügbar unter: https://de.statista.com/statistik/daten/studie/206071/umfrage/weltweite-co2-emissionen-nach-sektor/ (Zugriff am: 18. Februar 2022.296Z).

[5] Morgan Stanley, To Save the Planet, Start at Home | Morgan Stanley. [Online]. Verfügbar unter: https://www.morganstanley.com/ideas/fight-against-climate-change-starts-at-home/ (Zugriff am: 17. Februar 2022).

[6] Morgan Stanley, To Save the Planet, Start at Home | Morgan Stanley. [Online]. Verfügbar unter: https://www.morganstanley.com/ideas/fight-against-climate-change-starts-at-home/ (Zugriff am: 17. Februar 2022.271Z).

[7] N. Balta-Ozkan, B. Boteler und O. Amerighi, „European smart home market development: Public views on technical and economic aspects across the United Kingdom, Germany and Italy“, Energy Research & Social Science, Jg. 3, S. 65–77, 2014, doi: 10.1016/j.erss.2014.07.007.

[8] T. Hargreaves, C. Wilson und R. Hauxwell-Baldwin, „Learning to live in a smart home“, Building Research & Information, Jg. 46, Nr. 1, S. 127–139, 2018, doi: 10.1080/09613218.2017.1286882.

[9] Furszyfer Del Rio, Dylan D., B. K. Sovacool und S. Griffiths, „Culture, energy and climate sustainability, and smart home technologies: A mixed methods comparison of four countries“, Energy and Climate Change, Jg. 2, S. 100035, 2021, doi: 10.1016/j.egycc.2021.100035.

[10] Nazir Malik Umar Mahmud Younus Javed, „FUTURE CHALLENGES IN CONTEXT-AWARE COMPUTING“ (eng), IADIS digital library, S. 306–310, 2007. [Online]. Verfügbar unter: http://www.iadis.net/dl/final_uploads/200712L013.pdf

[11] M. Alaa, A. A. Zaidan, B. B. Zaidan, M. Talal und M.L.M. Kiah, „A review of smart home applications based on Internet of Things“, Journal of Network and Computer Applications, Jg. 97, S. 48–65, 2017, doi: 10.1016/j.jnca.2017.08.017.

[12] Dave Evans, „The Internet of Things: How the Next Evolution of the Internet Is Changing Everything“, S. 1–11. [Online]. Verfügbar unter: https://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf

[13] Z. A. Almusaylim und N. Zaman, „A review on smart home present state and challenges: linked to context-awareness internet of things (IoT)“, Wireless Netw, Jg. 25, Nr. 6, S. 3193–3204, 2019, doi: 10.1007/S11276-018-1712-5.

[14] Souza, A. M., & Amazonas, J. R., „A novel smart home application using an internet of things middleware.“, S. 1–7. [Online]. Verfügbar unter: https://ieeexplore.ieee.org/document/6525253/metrics#metrics

[15] M. R. Hoque, M. H. Kabir, J.-H. Jang und S.-H. Yang, „Middleware aided context-aware service for smart home“ in 2014 International Symposium on Consumer Electronics (ICSE), JeJu Island, South Korea, 6/22/2014 - 6/25/2014, S. 1–2, doi: 10.1109/ISCE.2014.6884408.

[16] U. Mahmud, S. Hussain und S. Yang, „Power Profiling of Context Aware Systems: A Contemporary Analysis and Framework for Power Conservation“, Wireless Communications and Mobile Computing, Jg. 2018, Nr. 3, S. 1–15, 2018, doi: 10.1155/2018/1347967.

[17] D.-M. Han und J.-H. Lim, „Design and implementation of smart home energy management systems based on zigbee“, IEEE Trans. Consumer Electron., Jg. 56, Nr. 3, S. 1417–1425, 2010, doi: 10.1109/TCE.2010.5606278.

[18] N. Naji, M. R. Abid, D. Benhaddou und N. Krami, „Context-Aware Wireless Sensor Networks for Smart Building Energy Management System“, Information, Jg. 11, Nr. 11, S. 530, 2020, doi: 10.3390/info11110530.

[19] S. Chen et al., „Butler, Not Servant: A Human-Centric Smart Home Energy Management System“, IEEE Commun. Mag., Jg. 55, Nr. 2, S. 27–33, 2017, doi: 10.1109/MCOM.2017.1600699CM.

[20] D. Lillis, T. O'Sullivan, T. Holz, C. Muldoon, M. J. O'Grady und G. M. P. O'Hare, „Smart Home Energy Management“ in Premier reference source, Recent advances in ambient intelligence and context-aware computing, K. Curran, Hg., Hershey, PA, USA: Information Science Reference, 2015, S. 155–168, doi: 10.4018/978-1-4666-7284-0.ch010.

[21] C.-H. Lu, C.-L. Wu, M.-Y. Weng, W.-C. Chen und L.-C. Fu, „Context-Aware Energy Saving System With Multiple Comfort-Constrained Optimization in M2M-Based Home Environment“, IEEE Trans. Automat. Sci. Eng., Jg. 14, Nr. 3, S. 1400–1414, 2017, doi: 10.1109/TASE.2015.2440303.

[22] Y. Nam, S. Rho und B.-g. Lee, „Intelligent context-aware energy management using the incremental simultaneous method in future wireless sensor networks and computing systems“, J Wireless Com Network, Jg. 2013, Nr. 1, 2013, doi: 10.1186/1687-1499-2013-10.

[23] M.-Y. Weng, C.-L. Wu, C.-H. Lu, H.-W. Yeh und L.-C. Fu, „Context-aware home energy saving based on Energy-Prone Context“ in 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2012), Vilamoura-Algarve, Portugal, 10/7/2012 - 10/12/2012, S. 5233–5238, doi: 10.1109/IROS.2012.6385762.

[24] Y. Raja Vara Prasad und P. Rajalakshmi, „Context aware building energy management system with heterogeneous wireless network

architecture“ in 2013 6th Joint IFIP Wireless and Mobile Networking Conference (WMNC 2013), Dubai, 4/23/2013 - 4/25/2013, S. 1–8, doi: 10.1109/WMNC.2013.6548976.

[25] A. Bhatt und J. Patoliya, „Cost effective digitization of home appliances for home automation with low-power WiFi devices“ in 2016 2nd International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB), Chennai, India, 2/27/2016 - 2/28/2016, S. 643–648, doi: 10.1109/AEEICB.2016.7538368.

[26] L. Smirek, G. Zimmermann und M. Beigl, „Just a Smart Home or Your Smart Home – A Framework for Personalized User Interfaces Based on Eclipse Smart Home and Universal Remote Console“, Procedia Computer Science, Jg. 98, S. 107–116, 2016, doi: 10.1016/j.procs.2016.09.018.

[27] B. Setz, S. Graef, D. Ivanova, A. Tiessen und M. Aiello, „A Comparison of Open-Source Home Automation Systems“, IEEE Access, Jg. 9, S. 167332–167352, 2021, doi: 10.1109/ACCESS.2021.3136025.

[28] F. Heimgaertner, S. Hettich, O. Kohlbacher und M. Menth, „Scaling home automation to public buildings: A distributed multiuser setup for OpenHAB 2“ in 2017 Global Internet of Things Summit (GIoTS), Geneva, Switzerland, 6/6/2017 - 6/9/2017, S. 1–6, doi: 10.1109/GIOTS.2017.8016235.

[29] A. Brice, „Best of open source smart home: Home Assistant vs OpenHAB“, 28. Feb. 2018, 2018. [Online]. Verfügbar unter: https://smarthome.university/your-smart-home-platform-home-assistant-vs-openhab/#:~:text=Add%2Dons%20are%20handled%20equally,basic%20configuration%20with%20Home%20Assistant. Zugriff am: 18. Februar 2022.596Z.

[30] OpenHAB Community, OpenHAB Empowering the smart home. [Online]. Verfügbar unter: https://www.openhab.org/docs/configuration/rules-dsl.html (Zugriff am: 18. Februar 2022.698Z).

[31] A. Brice, „Best of open source smart home: Home Assistant vs OpenHAB“, 28. Feb. 2018, 2018. [Online]. Verfügbar unter: https://smarthome.university/your-smart-home-platform-home-assistant-vs-openhab/. Zugriff am: 18. Februar 2022.946Z.

[32] Hass object | Home Assistant Developer Docs. [Online]. Verfügbar unter: https://developers.home-assistant.io/docs/dev_101_hass/ (Zugriff am: 18. Februar 2022.380Z).

[33] H. Assistant, Automating Home Assistant. [Online]. Verfügbar unter: https://www.home-assistant.io/getting-started/automation/ (Zugriff am: 18. Februar 2022.668Z).

Table of Figures