Towards Natural Interaction with Smart Homes

Francesco Benzi
University of Brescia
Via Branze 38, Brescia
Italy
f.benzi@unibs.it

Daniela Fogli
University of Brescia
Via Branze 38, Brescia
Italy
daniela.fogli@unibs.it

Giovanni Guida
University of Brescia
Via Branze 38, Brescia
Italy
giovanni.guida@unibs.it

ABSTRACT

This paper proposes a novel conceptual architecture for smart homes as well as a new approach to home-inhabitant interaction. HOME is an intelligent agent based on a Belief-Desire-Intention (BDI) model, extended with the concept of values that represent abstract and permanent needs, such as health, comfort or security, which drive its decision making activity. Values, beliefs, desires and intentions are also the basic conceptual entities of inhabitant-system dialogue: the ultimate goal of HOME is to satisfy inhabitant needs and preferences, by taking into account knowledge about the environment and available devices, such as sensors and actuators. The result is a peer-to-peer interaction, where (the best) decisions emerge from a dialectic behavior.

CCS CONCEPTS

• **Human-centered computing~Ambient intelligence** • Human-centered computing~HCI theory, concepts and models

KEYWORDS

Smart home, intelligent autonomous agent, BDI architecture, natural interaction, value-based reasoning

ACM Reference format:

F. Benzi, D. Fogli, G. Guida. 2017. Towards Natural Interaction with Smart Homes. In *Proceedings of ACM 12th Biannual Conference of The Italian SIGCHI Chapter, Cagliari, Italy, September 2017 (CHITALY 2017)*, 6 pages.
https://doi.org/10.1145/3125571.3125581

1 INTRODUCTION

Ambient Intelligence (AmI) is a discipline whose goal is to make our living environments intelligent and able to support us unobtrusively [22]. Traditional research in AmI mainly focuses on hardware devices, the so-called Internet of Things, and on artificial intelligence (AI) techniques aimed at endowing an environment with intelligent behaviors, such as reasoning capabilities [3] or learning features [2]. Agent-based architectures for AmI have been proposed as well (for example [23-24]), where an agent is considered as "anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors" [21]. The home is usually chosen as the paradigmatic environment in AmI, and thus several literature proposals focus on the so-called "smart home" [14][19][23].

Human-computer interaction (HCI) researchers, on the other hand, focus their attention to AmI systems that can be easily managed and controlled by the end users, for example through smart interfaces that support end-user programming to define the desired system behavior [13].

AI and HCI perspectives appear to pursue opposite goals in AmI: the former aims to create a system that pro-actively assists the user in his/her daily activity and learns by observing patterns of behaviors; the latter focuses on user interfaces that facilitate smart home control by its inhabitants. However, both intelligent and user-controllable behaviors should co-exist in a future generation smart home. In this paper, we propose a novel conceptual architecture that aims at integrating these two perspectives. We regard the home and the user as two agents able to "naturally interact" through a common language, based on mental attitudes, namely beliefs, desires, intentions, and values.

In particular, a smart home is herewith conceived as an intelligent and autonomous software agent, called HOME (Home Operation & Management Environment), whose objective is to support the user in his/her daily activities inside a house. Belief-desire-intention (BDI) theory [4][18] – extended with the new concept of value – is adopted both to model smart home behavior and to express users' needs and requests to HOME. The final purpose is to obtain a balance between the intelligent proactive behavior of a home and the user control over it [12]. Therefore, we regard an AmI solution as a socio-technical system, where user and HOME cooperate to make the environment helpful, comfortable and adaptable over time [6].

2 RELATED WORK

As already mentioned, in HCI literature smart home control has been studied from an end-user programming perspective, in

CHItaly '17, September 18–20, 2017, Cagliari, Italy

ACM ISBN 978-1-4503-5237-6/17/09…$15.00
https://doi.org/10.1145/3125571.3125581

order to design user interfaces supporting the definition of home behaviors. Event-condition-action (ECA) rules are at the basis of most of the proposed approaches (among others, [8][10-11][15][17][25]). For instance, AppsGate, presented in [8], is a software environment that combines rule-based and imperative programming through a visual language, which however requires that users are properly trained and invest time and effort to program home behaviors.

Other research scholars in HCI study natural interaction with smart homes with reference to natural language voice interfaces. In [14], a central home controller provides a wide range of possible sentences to control the home and an avatar representation is able to answer with synthetic voice. However, this project mainly aims at offering an alternative interface, without providing any kind of reasoning on users' requests. Clark et al. [7] emphasizes the need of AI and machine learning in supporting advanced interaction features. As an example in this direction, reinforcement learning is used in [5] to implement a speech recognition application that learns to understand and contextualize the queries of the user, in order to execute the right action on the home. Several and meaningful interaction instances are however necessary as the input for the learning algorithm.

Agent-based architectures represent the most frequent proposal in AI to address the problem of Ambient Intelligence. Tapia et al. specify that "the agents have characteristics such as autonomy, reasoning, reactivity, social abilities and pro-activity which make them appropriate for developing dynamic and distributed systems based on AmI" [24]. For instance, the project MavHome [26] conceives the environment as an intelligent agent able to collect perceptions from sensors and to identify repetitive patterns. In [20], instead, the agent-oriented AmI system integrates ideas inspired by Mirror Worlds [16], namely a digital world mirroring but also augmenting the physical world with new capabilities, services and functionalities. Few literature proposals conceive AmI systems (smart homes in particular) as autonomous intelligent agents whose behavior is determined by three mental attitudes, called beliefs, desires and intentions (BDI). Among them, [19] chooses to adopt BDI deliberative agents instead of simple reactive agents, since they appear to be better suited to tackle AmI challenges, such as non-determinism, mutually-exclusive objectives, context-dependence, limited time for computations and actions. Sun et al. [23] propose a multi-agent system where each single agent is modeled as a BDI agent, and multi-agent group behavior is controlled by a set of regulation policies.

In this paper, the smart home is conceived as a V-BDI agent, i.e. a traditional BDI agent with an added overarching element, called values (V). Inspired to value theory in sociology and economics [9], values represent abstract and permanent needs of the user that the system should satisfy, like for example security, comfort, savings, or health. Their role is to modulate the behavior of the other three mental attitudes, according to the hierarchy of values of the home inhabitants. Similarly to other proposals, our architecture includes a rule-based mechanism to support agent reasoning on the basis of beliefs, desires, intentions and values. However, in our proposal, rules are not directly related to the control of smart accessories available in the home as in [8][10][15][25], but they are aimed at governing the general reasoning and decision-making patterns that characterize the operation of HOME.

3 USER-HOME INTERACTION SCENARIO

A hypothetical interaction scenario between Bruno, an hypothetical home inhabitant, and HOME is described below. It will be taken as reference to illustrate the approach and the novel architecture proposed in the paper.

"It's a hot summer afternoon and to keep Bruno's house comfortable but also to save energy costs, the temperature is kept around 25°C. Bruno is leaving his office and as soon as he enters his car, he calls HOME to communicate that he is arriving. HOME realizes that the temperature must be brought to 21°C, to prepare a more confortable environment for Bruno. According to Bruno's preferences, comfort is regarded as more important than energy savings. When Bruno arrives at home, HOME welcomes him by telling: "Hi Bruno, how are you?". Bruno replies "Huh, it's cold here!". HOME knows that a transition from a very hot environment to a colder one may provide the sensation of a too cold environment and thus it answers as follows: "It's very hot outside, I prepared the house with a confortable temperature, please wait some minutes". After a while, HOME asks: "Is it ok now?" Bruno confirms that temperature is ok. HOME would like to be helpful and asks again: "Are you tired, Bruno?" Bruno replies: "Yes, I am tired, what do you propose?" HOME knows that Bruno will soon have dinner and that, when he is tired, he prefers ordering a pizza and not cooking at all. It also knows that eating a pizza on the sofa could be done while watching a movie on TV. Therefore, HOME proposes a relax situation: "We can order a pizza and watch TV". But Bruno replies: "I already ate pizza at lunch" and "I would like a healthier dinner!" Home knows that a healthy behavior means that you must eat different things in a day, including vegetables and fruit. Therefore, it looks for fast recipes including the ingredients that are present in the fridge and larder and proposes Bruno to cook one of them."

4 MODELING USER-HOME INTERACTION THROUGH V-BDI

Values are abstract and permanent needs of the user that the system should satisfy, like for example security, comfort or health. In the system, values are ordered on the basis of the active context and on what the user values most. According to [1], a *context* is "any information that can be used to characterize the situation of an entity". More precisely, it is intended here as a situational pattern that captures a recurring situation related to the home and its inhabitants. For instance, at the beginning of the above scenario the home is recognized as empty; therefore, the active context is "Nobody_at_home". This context is associated with the following ordered list of values: 1) security, 2) energy saving, 3) comfort, 4) health.

According to [4], *beliefs* represent the information an agent has about the world it inhabits, and about its own internal state. Beliefs can be generated by the system through reasoning about perceptions on the environment and on the basis of knowledge about users' preferences. For instance, given the scenario above, at the beginning the system has the following beliefs:

S_bel_1: Home(empty)
S_bel_2: Season(summer)
S_bel_3: Inside_preferred_temperature(21°C)
S_bel_4: Inside_suggested_temperature(25°C)
S_bel_5: Inside_temperature(27°C, over_suggested)

The context "Nobody_at_home" previously mentioned is activated as a consequence of belief S_bel_1.

Desires are environment states that the system tries to bring about. Desires strictly depend on the changing needs of the owner of the environment and on current beliefs, i.e. desires change depending on context. Desires can be of two types: *single run*, i.e. once the desired environment state has been reached the desire is satisfied and can be deleted from the current desire list (this is for example the case of sending a message, asking a question to the user, or closing a window), and permanent, i.e. the system must constantly generate actions to keep the desired condition true over time (this is for example the case of maintaining a specified temperature in a room). Both types of desires are not deleted when their associated context is no more active, but are retracted if they lose a conflict with another desire.

The context "Nobody_at_home" is associated to the desire

S_des_1: Keep_suggested_temperature

that aims to satisfy value "energy saving".

Intentions represent the current course of action the system has selected to satisfy a desire, i.e. sequences of actions on the environment used to reach the state specified in a desire. In the example, the intention selected to satisfy S_des_1 will try to keep temperature around the suggested value by means of the air conditioner, since outside temperature is higher than inside temperature. Therefore, we will have

S_int_1: Keep(Inside_suggested_temperature(25°C), Air_conditioner)

which in turns brings to action Set(Air_conditioner, 25°C).

Also the user may communicate to the system his/her own beliefs. For instance, when Bruno communicates HOME that he is arriving a new user belief is created:

U_bel_1: Here_soon(Bruno)

which triggers the context: "Bruno_arriving". This new active context is associated to a different ordered list of values (where "comfort" has higher priority that "energy saving") and brings about a new desire

S_des_2: Keep_preferred_temperature

that aims to satisfy value "comfort". A conflict thus arises between S_des_1 and S_des_2 because the states they would like to reach are contradicting (temperature equal to 25°C vs. temperature equal to 21°C). S_des_2 wins the conflict since it is associated with a higher priority value. So intention S_int_1 is retracted and S_int_2 is brought about to keep inside temperature around that preferred by Bruno.

When Bruno arrives home, S_bel_1, S_bel_5 and U_bel_1 are retracted, while new system beliefs are created

S_bel_6: Inside_temperature(21°C, preferred)
S_bel_7: Home(Bruno_here)
S_bel_8: Just_arrived(Bruno)
S_bel_9: Transition(Bruno, hot, cold)

The new belief base determines a context change: both "Nobody_at_home" and "Bruno_arriving" are deleted, while "Bruno_at_home" and "Welcome_Bruno" are activated. Desire S_des_2 persists, since it is not in conflict with the new contexts. However, when Bruno says "Huh, it's cold here!", a new user belief is created, as follows:

U_bel_2: Inside_temperature(21°C, under_preferred)

This belief is in conflict with S_bel_6, but the combination of S_bel_6 and U_bel_2 with S_bel_9 determines the recognition of a new active context "Transition_hot_cold" which brings about a new desire:

S_des_3: Explain_situation

This desire generates two intentions:

S_int_3: Explanation(Transition_hot_cold)
S_int_4: Ask(Transition_ended)

Therefore, the actions of telling Bruno to wait for some minutes and then asking again if the temperature has become acceptable are executed; at the end, U_bel_2 is retracted and the context "Transition_hot_cold" is deleted (of course the context "Welcome_Bruno" remains active).

According to the currently active context, HOME asks the user: "Are you tired, Bruno?" As a consequence of Bruno's reply a new user belief is created:

U_bel_3: Bruno(tired)

This belief has the effect of changing the active context from "Welcome_Bruno" into "Relax_at_home" with the corresponding list of ordered values: 1) comfort, 2) health, 3) security, 4) energy saving. As a consequence, new desires will be brought about:

S_des_4: Fast_dinner
S_des_5: Sofa_TV_watching

Both desires are associated with value "comfort".

Desire S_des_4 has the following associated intentions:

S_int_5: Propose(pizza)
S_int_6: Order(pizza)

Intention S_int_5 brings to Action: Tell("We can order a pizza and watch TV"). But when Bruno replies that he already ate pizza at lunch, a new user belief is asserted:

U_bel_4: Eat(pizza, lunch)

Furthermore, when Bruno adds that he wants to have a healthier dinner, the following user desire is created:

U_des_1: Healthy_eating

Given U_bel_3 and U_Bel_4, the new context "Eat_healthy_and_fast" is activated, thus changing the current ordered list of values into: 1) health, 2) comfort, 3) security, 4) energy saving. Desire U_des_1 generates the intention

S_int_7: Propose(healthy_dinner)

which is however in conflict with S_int_6. Considering the current ordered list of values, intention S_int_7 is preferred to S_int_6. So in the end S_int_7 brings to action Propose_healthy_recipe(dinner, easy, fast).

5 CONCEPTUAL ARCHITECTURE

Figure 1 illustrates the architecture of HOME at a high level of abstraction. The *Belief Generation Module* (*BGM*) is responsible of gathering sensor data and user utterances and generating system and user beliefs, by instantiating suitable *belief types*. The BGM is constituted by a collection of belief generators, each one responsible of instantiating a specific belief type based on sensor readings and user utterances. We assume that the user may express his/her own beliefs through a mobile application able to interpret voice commands and forward them to BGM. If conflicts occur between beliefs, BGM is in charge of solving them by means of coherence rules. The beliefs currently asserted are stored in a *Belief List*.

The *Context Generation Module* (*CGM*) is devoted to identify the current context to be applied to the management of desires and intentions. On the basis of the beliefs asserted in the *Belief List*, it selects a suitable *context type* and instantiates it to the current state of the world. Available context types are organized in a tree, where each node represents a specialization of its ancestor. Each context type is associated to a trigger, which defines the conditions for the context to be activated. CGM explores the tree through a depth-first approach, to identify the most specific context types that can be activated in the current situation. In general, more than one context can be active at any moment; an *idle context* is active when no other context is active. As an example, Figure 2 shows the context tree created in the interaction scenario modeled in the previous section. The user may explicitly activate a context for example through a voice command (for example: "begin cooking" or "going to bed"). Each context is associated with an ordered list of values. The new active context generated by CGM is passed to the modules devoted to manage desires and intentions.

The *Desire Generation Module* (*DGM*) is responsible of the generation of desires by instantiating suitable *desire types*. Based on the currently active contexts and on the current belief list several desires may be instantiated. Each desire is associated with a list of the values it promotes and may specify possible precedence relations with other desires. DGM exploits priorities between values to solve conflicts between desires and precedence relations between desires to correctly order the desires on the time axis.

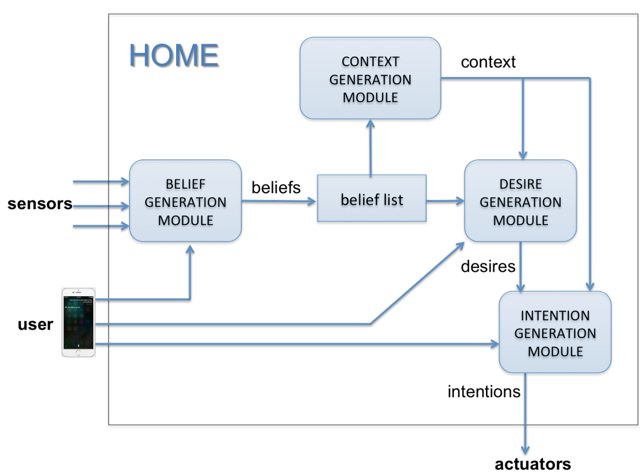

Figure 1: Conceptual architecture of the HOME application.

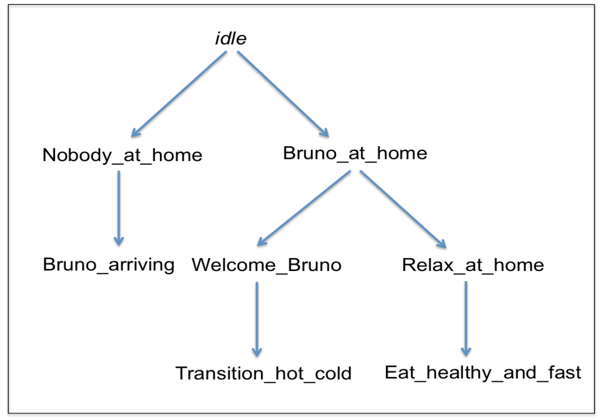

Figure 2: Example of context tree.

The user can interact with DGM (but not definitely control it) by directly expressing a desire (for example: "I want more light" or "I would like healthy eating"). Desires expressed by the user may conflict with system-generated ones and, once again, conflicts are solved according to the priorities of the values they promote. In summary, DGM produces an ordered and consistent list of desires, which is passed to the module devoted to manage intentions.

The *Intention Generation Module (IGM)* is responsible of deciding which actions to execute given the current list of desires. For each desire, a set of alternative intentions may be generated, through instantiation of *intention types*. The user may express his/her personal intentions (for example: "Order a pizza" or "Lower temperature"), which will be taken into consideration by IGM. Each intention is associated with a list of values it promotes or demotes and may specify possible precedence relations with other intentions. Values are used to choose the most appropriate intentions to achieve a given desire and precedence relations are exploited to order intentions in the correct way. IGM produces an ordered and consistent list of intentions, which are eventually executed by suitable actuators.

All reasoning modules are conceived as rule-based systems exploiting Event-Condition-Action (ECA) rules. Inputs to a module will trigger one or more rules relevant to that module by matching events and conditions; the action part of the rules will instead include the output generated by the module. Default rules for the most typical contexts, beliefs, desires and intentions are defined at system design time and are customized by the technician in charge of system set up according to the habits and preferences of home inhabitants collected through a focused interview. Then, at run time, an innovative user interface supports end users in rule modification.

6 USER INTERFACE

The user can engage HOME in a dialogue or be engaged in a dialogue by HOME in any moment during system operation. In particular, he/she can provide information that will result into user-generated beliefs and can formulate queries or commands that give rise to user-generated desires or intentions, or can even change the ordering of values defined at design time. More importantly, however, the interaction with the user can allow the system to update the content of its knowledge bases, by modifying existing rules for example by resetting parameters, dropping rules that turn out to be outdated or invalid, adding new rules necessary to face new situations. To this purpose, differently from most approaches in literature [8][10-11][15][17][25], we do not assume to support the user with a graceful rule editor or to ask him/her directly the pieces of information necessary to compile rules. Our standpoint is to enable the system to derive the information items necessary for rule updating from a natural dialogue with the user in a non-intrusive and non-inquisitive way. The dialogue may be started by the user or by the system, since HOME is instructed to take the initiative if the user has a too passive behavior.

Through the dialogue, the user may express his/her assessment about HOME performance. A positive feedback provides HOME with useful information to corroborate the validity of the current content of the knowledge bases; a negative feedback, followed by a more focused dialogue, can be the source of information necessary for rule updating. The dialogue can also be the right tool for the user to inform the system about changed habits, constraints or preferences, thus allowing it to reshape its behavior accordingly. In any case, the user-system dialogue does not directly concerns the rule base, but is assumed to be at a higher level, closer to the user world and far away from the technicalities of rule editing.

We assume that the user-system dialogue is based on both visual interaction and natural language, and can occur on a mobile application by exploiting voice recognition. Aspects related to natural language understanding are however outside the scope of this paper. As an example, Figure 3 shows the mock-up of the visual interface allowing Bruno to communicate HOME that his preferences about how to achieve the desire "Fast dinner" have changed. In particular, the dialogue starts from screenshot in Figure 3(a), where Bruno may choose between two options: "GIVE FEEDBACK about HOME performance" and "INFORM HOME about new preferences or constraints". The user selects the second one, and HOME in turn asks to indicate the situation affected by the new preferences or constraints (Fig. 3(b)). The user selects the "Relax" situation and then chooses "Fast dinner" among available options for a relaxing evening. "Fast dinner" is a desire that can be achieved through different lists of intentions: in particular the user wants to change preferences associated to "Order pizza" (Fig. 3(c)), by telling HOME that pizza can be ordered only in the weekend (Fig. 3(d)).

7 CONCLUSION

This paper has presented the conceptual architecture of a smart home fostering a natural user-system interaction. We are currently implementing HOME through JADEX, a framework for the development of agent-based systems [4]. We are also developing the mobile application supporting user-HOME interaction.

Figure 3: **Mock-ups of the user interface.**

This first version will implement only a basic set of features, sufficient for supporting a smooth system behavior. Among more advanced issues that will be faced in the future we mention:

- the control of the depth-first search in the context tree, that is how to decide when to stop search (the first matching context, the first deepest matching context, all deepest matching contexts);
- the management of persistency of values, beliefs, desires and intentions over time, that is when or how frequently their validity needs to be verified and eventually updated;
- the strategy of the rule interpreter, that is the logical relation between trigger and condition (now assumed to be in a basic logical *and* relation), the possibility to have complex conditions (built with *and*, *or* and *not* logical connectives), the possibility to have composite actions including more items whose execution can be controlled by means of a suitable temporal logic.

REFERENCES

[1] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, P. Steggles. 1999. Towards a better understanding of context and context-awareness. *Proc. 1st International Symposium on Handheld and Ubiquitous Compting (HUC'99).* London, UK: Springer-Verlag, 304-307.

[2] A. Aztiria, A. Izaguirre, J. C. Augusto. 2010. Learning patterns in ambient intelligence environments: a survey. *Artif. Intell. Rev.* 34, 1, 35-51.

[3] A. Bikakis, G. Antoniou. 2010. Rule-Based Contextual Reasoning in Ambient Intelligence. In M. Dean, J. Hall, A. Rotolo & S. Tabet (Eds.), *Semantic Web Rules* (Vol. 6403, pp. 74-88). Springer Berlin Heidelberg.

[4] L. Braubach, W. Lamersdorf, A. Pokahr. 2003. Jadex: Implementing a BDI-infrastructure for JADE agents. *Exp*, 3(3), 76-85.

[5] A. Brenon, F. Portet, M. Vacher. 2016. Preliminary Study of Adaptive Decision-Making System for Vocal Command in Smart Home. *Proc. 12th International Conference on Intelligent Environments (IE 2016),* 218-221.

[6] F. Cabitza, D. Fogli, A. Piccinno. 2014. Fostering participation and co-evolution in sentient multimedia systems. *Journal of Visual Languages and Computing*, vol. 25(6), 684-694.

[7] M. Clark, P. Dutta, M. W. Newman. 2016. Towards a natural language programming interface for smart homes. *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct.* ACM Press.

[8] J. Coutaz, J. L. Crowley, 2016. A First-Person Experience with End-User Development for Smart Homes, *IEEE Pervasive Computing*, April-June, 26-39.

[9] Gerard Debreu, 1972. *Theory of Value: An Axiomatic Analysis of Economic Equilibrium.* New Haven and London, Yale University Press.

[10] G. Desolda, C. Ardito, M. Matera. 2017. Empowering End Users to Customize their Smart Environments: Model, Composition Paradigms, and Domain-Specific Tools. *ACM Trans. Comput.-Hum. Interact.* 24(2), 12:1-12:52.

[11] Y. Dahl, R.-M. Svendsen. 2011. End-User Composition Interfaces for Smart Environments: A Preliminary Study of Usability Factors. In A. Marcus (Ed.), *Design, User Experience, and Usability. Theory, Methods, Tools and Practice* (Vol. 6770, pp. 118-127). Berlin Heidelberg: Springer.

[12] S. Davidoff, M. K. Lee, C. Yiu, J. Zimmerman, Anind K. Dey, 2006. Principles of Smart Home Control. In P. Dourish & A. Friday (Eds.), *UbiComp 2006: Ubiquitous Computing* (Vol. 4206, pp. 19-34). Berlin Heidelberg: Springer.

[13] D. Fogli, R. Lanzilotti, A. Piccinno, 2016. End-User Development Tools for the Smart Home: A Systematic Literature Review, In: N. Streitz and P. Markopoulos (Eds.): *DAPI 2016*, LNCS 9749, Springer International Publishing Switzerland, 2016, 1–11.

[14] A. Gárate, N. Herrasti, A. López. 2005. GENIO: an ambient intelligence application in home automation and entertainment environment. *Proceedings of the 2005 joint conference on Smart objects and ambient intelligence: innovative context-aware services: usages and technologies.* ACM Press.

[15] M. García-Herranz, P. A. Haya, X. Alamán, 2010. Towards a Ubiquitous End-User Programming System for Smart Spaces. *Journal of Universal Computer Science*, 16(12), 1633—1649.

[16] David Gelernter. 1993. *Mirror Worlds: Or the Day Software Puts the Universe in a Shoebox... How It Will Happen and What It Will Mean.* Oxford 1993.

[17] G. Ghiani, M. Manca, F. Paternò, C. Santoro. 2017. Personalization of Context-Dependent Applications Through Trigger-Action Rules. *ACM Trans. Comput.-Hum. Interact.* 24(2), 14:1-14:33.

[18] A. S., Rao, M. P. Georgeff. 1995. BDI agents: From theory to practice. *Proc. ICMAS.*

[19] C. Reinisch, W. Kastner. 2011. Agent based control in the smart home. *Proc. IECON 2011-37th Annual Conference on IEEE Industrial Electronics Society.* IEEE press.

[20] A. Ricci, M. Piunti, L. Tummolini, L. 2015. The Mirror World: Preparing for Mixed-Reality Living. *IEEE Pervasive Computing*, 14(2), 60-63.

[21] S. Russell, P. Norvig. 1995. *Artificial Intelligence - A modern approach.* Prentice-Hall.

[22] Fariba Sadri 2011. Ambient intelligence: A survey. *ACM Computing Surveys,* 43(4), 1-66.

[23] Q. Sun, W. Yu, N. Kochurov, Q. Hao, F. Hu. 2013. A Multi-Agent-Based Intelligent Sensor and Actuator Network Design for Smart House and Home Automation. *J. Sens. Actuator Netw.* 2(3), 557-588.

[24] D. I. Tapia, A. Abraham, J. M. Corchado, Ricardo S. Alonso. Agents and ambient intelligence: case studies. 2010. *Journal Ambient Intell Human Comput,* 1(2), 85-93.

[25] B. Ur, E. McManus, M. P. Y. Ho, M. L. Littman. 2014. Practical trigger-action programming in the smart home. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '14). ACM, New York, NY, USA, 803-812.

[26] G. M. Youngblood, D. J. Cook. 2007. Data Mining for Hierarchical Model Creation. *IEEE Transactions on Systems, Man, and Cybernetics, Part C,* 37(4), 561-572.