Towards Understanding the Transparency of Automations in Daily Environments

Fabio Paternò, Simone Gallo, Marco Manca, Andrea Mattioli, Carmen Santoro

CNR-ISTI, HIIS Laboratory, Pisa, Italy

Abstract

This paper outlines a proposal for how to address transparency of automations in daily environments, such as smart homes, based on experiences carried out in previous projects. The trigger-action programming paradigm has been used to describe and implement such automations in both commercial and research tools. Such automations can be generated through machine learning techniques or directly by the end users or through an interaction between an intelligent agent and the user. When they are executed the resulting behaviour does not always result in the desired actions, and users may have difficulties in understanding and controlling them. Thus, there is a need for design criteria and associated tools that help people to understand and control what happens with the automations active in the environments where they live, and explain how they work and can be modified to better meet their needs.

Keywords

End-user development, Everyday automation, Internet of Things

1. Introduction

How people interact with digital technologies is currently caught between the Internet of Things (IoT), where objects are continuously increasing their technological capabilities in terms of functionalities and connectivity, and Artificial Intelligence, which is penetrating many areas of daily life by supporting their increasing ability to autonomously activate functionalities based on collected data and statistically-based forecasts. In both trends, human control over technology is jeopardized, little is happening in terms of innovating how we think and control automations.

We live more and more in environments with dynamic sets of objects, devices, services, people, and intelligent support. This opens up great opportunities, new possibilities, but there are also risks and new problems. The available automations can be created through machine learning techniques [18, 21] and activated or recommended [15, 18] to users, or can even be directly created by them. Trigger-action programming [8, 19] has often been used to describe and implement automations in environments rich in terms of presence of connected objects, devices, and services. It is based on sets of rules that connect the dynamic events and/or conditions with the expected reactions without requiring the use of complex programming structures, and it has been used in several domains, such as home automation [1, 16, 19], ambient assisted living [14], robots, [11], finance [6]. However, when they are automatically generated some problems can occur if the end user's viewpoint is not sufficiently considered. For example, the study reported in [20] describes how a learning system can fail to adapt to recent user changes or the difficulty users have understanding what information the system requires in order to be trained to generate the desired behaviour. Likewise, a survey-based study with participants who have smart devices in their own home [9], reported difficulties in avoiding false alarms, communicating complex schedules, and resolving conflicting preferences. Such issues highlight the importance of providing

IUI Workshops, March 2022, Helsinki, Finland
EMAIL: {fabio.paterno, simone.gallo, marco.manca, andrea.mattioli, carmen.santoro}@isti.cnr.it

© 2020 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

conceptual and technological support for improving the transparency of such automations. Thus, there is a need for novel solutions able to support what we refer to as "**humanations**", which are automations that users can understand and modify.

2. Conceptual Dimensions

We can better address automation transparency if we identify the set of dimensions that can characterise this concept. Design spaces for understanding automations have been proposed in previous work [3, 17] but we find that design criteria for their transparency have not been sufficiently addressed. For this purpose, the first important point to clarify concerns the possible desired **levels of user control**. We can identify at least four possible levels: perception (users are able to perceive that some automation is active and working), understanding (users are able to understand how such automation works, thus some level of explainability is supported), predictability (users are able to foresee what will happen in the future with the current active automation), modification (users are enabled to change something in the automations when their results are not satisfying).

For example, we can consider a smart home where the heating system is automatically activated when the user is at home and the temperature is below 17 Celsius degrees and the time is after 5 pm. The first level of control indicates that the user is able to detect that some evenings the heating system is sometimes activated automatically (automation perception). In order to ensure that users understand an automation it is necessary that they be able to know what elements are necessary to trigger the automation (in this example, user location, temperature, and time), when they actually trigger the rule, and what the corresponding action is. Predictability is achieved when the user is able to understand the future behaviour of the smart home [5]. Thus, for example, the user is able to indicate whether the heating system will be active or not at a given time (e.g. 4 pm). Lastly, if the user is able to modify such automation, for example for activating the heating system at different time and with different temperature, then the automation modification level is reached.

Another relevant dimension is the **granularity** of the set of objects involved in the considered automations. The focus can consider a single object, which for some reason is of interest for the user. For example, there is a lamp in front of the user who may be interested in the automations that control it. The focus can also consider a group of objects (e.g. the lamps that are nearby) or can be more general and consider all the connected objects that are in a given space (e.g. in a room or a an entire flat).

One further dimension is represented by **the temporal aspects** of automations [4, 10], which can be composed of triggers and actions, both of which have different temporal aspects. Triggers can be composed of events and conditions, where events are instantaneous changes in some contextual element, while conditions are associated with the state of some elements, which can last for some time. Likewise, the effects of the actions can be instantaneous (e.g. sending a notification) or can have longer duration (e.g. turn a light on). Thus, the combination of triggers and actions can determine different types of situations depending on the temporal aspects of the constituent elements, which should be clearly expressed to allow users to fully understand and eventually modify the automations of interest.

One further aspect to consider for automation transparency is their **analytics** in other words support for analysing the data on how they have been used. Automations go through three stages: creation, enabled and execution. Regarding their creation it is interesting to know what agent created them and when. Then, it can be useful to know the periods of time when they have been enabled, meaning executable. Another aspect of interest in their use is when and how many times they have been executed. This information is also useful to understand whether the automation is working as expected or it is executed at the wrong times or there are some correlations between them and specific contexts of use.

3. Tool Support

If we want to provide tool support for the transparency of daily automations we need to think about something that can be used frequently in many locations and situations, with limited effort. In addition, it should be something through which we can immediately interact with the variety of connected objects and sensors that may be involved in the automations. For this purpose, we can consider two possible directions. One is the use of conversational agents, where users can ask in natural language what the current automations are, why they are active, and modify them, if not completely satisfactory by using devices such as Alexa or Google Home or their smartphone [7]. Another possibility is an augmented reality smartphone-based application, which seems a relevant direction to investigate since the smartphone is the device that people most often have with them, and it is immediate for them to frame the surrounding objects of interest to receive relevant information through its camera. Augmented Reality is a technology that nowadays has reached a widespread application in many domains for its ability to connect virtual and physical elements. However, so far, in IoT applications, it has mainly been used to superimpose digital information about smart objects available in the current user context, primarily concerning their state and capabilities [1]. We need to better exploit this technology to support automation transparency, in order to make the intelligence at work in the surrounding environment perceivable, so that users can know what automations involving the nearby objects are active, and modify them, if necessary.

Regarding the levels of user control, relevant solutions should be able to highlight whether the surrounding objects shown in the smartphone screen are involved in active automations. They should be able to explain what automations are active on request, and also allow users to modify them, even providing suggestions, if they do not meet their needs. In order to support the granularity dimension, the tool should be able to provide information not only of the automations involving a single framed object but also those related to groups of objects, for example a group graphically selected in the smartphone camera supported view, or the entire current environment where the user is located (e.g. a kitchen). This implies that the solution include a connection with some indoor localization technology.

To support the temporal dimension one key aspect is to provide explicit indications whether the elements composing the trigger side are events or conditions. For this purpose, it is possible to use different keywords (e.g. “when” for events, “if” or “while” for conditions). One further support is to avoid the creation of automations whose components contain erroneous temporal relations. For example, a trigger defined by the composition of two events with an AND logical operator is almost impossible to occur since it is very unlikely that the two events occur at the same time. Another example of a problematic situation is when the trigger is a condition and the action is instantaneous. Since the condition can last for some time, when should the action be performed? Since we can assume that the instantaneous actions should be performed only once, then the trigger should instead indicate an event to identify when it is to be performed.

One initial possible solution addressing such aspects has been proposed in [2] with the SAC app.

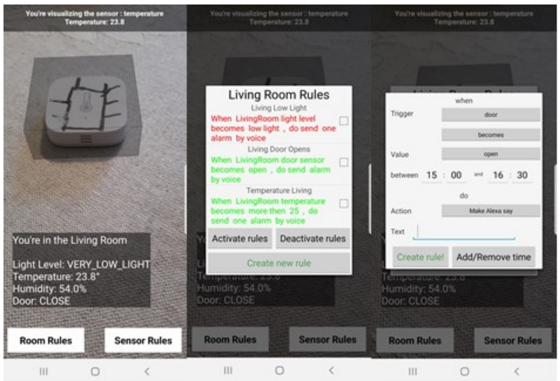

Figure 1: The SAC app (from [2])

Figure 1 shows the types of interactions and representations that it supports: (left) info on the current room (Living Room) and the framed sensor; (centre) the rules created for the current room; (right) the support for creating new rules.A first user study gathered positive feedback, but in order to fully support transparency, a richer set of information should be provided, and also augmented reality can be better exploited. The Vuforia functionalities were used to support object recognition. They worked sufficiently well but in some cases the sensors had to be manually marked to facilitate their recognition (see an example in Figure 1, left), and users had to be sufficiently close, with the focus of the camera on them for some seconds in order to perform their recognition. Thus, a solution based on a computer vision technique exploiting Convolutional Neural Networks can be more efficient, if adequately trained.

Another relevant experience has been carried out in the AAL PETAL project, where a prototype platform (TAREME) has been designed and developed for supporting caregiver management of automations in the homes of older adults with mild cognitive impairments in order to provide personalised support in their daily activities. In order to allow caregivers to better understand the automations, the tool was extended [13] to allow them to indicate a possible context of use and some automations, and then it provided feedback on what automation would have been triggered in that context, with the possibility to receive and explanation in natural language on why or why not they would have been executed. The platform also includes functionalities for remote monitoring and analytics of the automations [14]. Figure 2 shows some of the information that it is able to display.

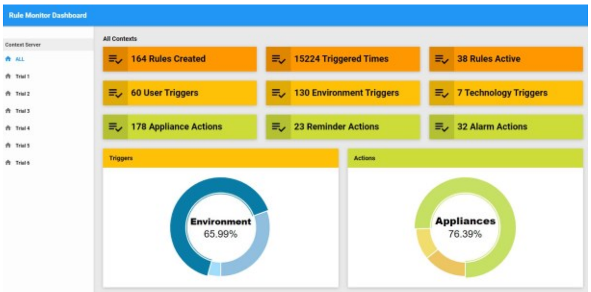

Figure 2: The TAREME display of some automation analytics (from [14])

The platform is able to monitor automations from multiple sites at the same time. In the example reported in the figure there are six trial sites active, and it shows on the top the total number of rules created, how many times they have been triggered and how many are currently active. The tool also supports the possibility to filter the displayed information only for one specific site. In addition, as the figure shows, the tool also categorizes the triggers depending on whether they are related to the user behaviour, environmental aspects or some device, and indicates how many triggers belong to each category. Likewise, also the numbers of associated actions are displayed classified depending on whether they are performed on some appliance or they are reminders or alarms.

4. Conclusions

In this paper we introduce the concept of transparency of automations in daily environments, and some logical dimensions that characterise it. Such dimensions are provided at a conceptual design level, and we also report and discuss how we have addressed them with some tools in previous projects.

Future work will be dedicated to extending and validating the identified design aspects, and provide improved associated tool support, for example with more thorough treatment of explainability aspects [9].

5. Acknowledgments

Support from the PRIN EMPATHY (http://www.empathy-project.eu/) project is gratefully acknowledged

6. References

[1] Aghaee, S. and Blackwell, A.F. (2015). IoT programming needs deixis. Proceedings of CHI 2015 Workshop on End User Development in the Internet of Things Era - EUDITE)

[2] R. Ariano, M. Manca, F. Paternò, C. Santoro, Smartphone-based Augmented Reality for End-User Creation of Home Automations, Behaviour & Information Technology, 2022, https://www.tandfonline.com/doi/full/10.1080/0144929X.2021.2017482

[3] J. Bongard, M. Baldauf, and P. Fröhlich. 2020. Grasping Everyday Automation – A Design Space for Ubiquitous Automated Systems. In 19th International Conference on Mobile and Ubiquitous Multimedia (MUM 2020), November 22–25, 2020, Essen, Germany. ACM,

[4] Will Brackenbury, Abhimanyu Deora, Jillian Ritchey, Jason Vallee, Weijia He, Guan Wang, Michael L. Littman, Blase Ur: How Users Interpret Bugs in Trigger-Action Programming. CHI 2019: 552

[5] S. Coppers, D. Vanacken, K. Luyten, FORTNIoT: Intelligible Predictions to Improve User Understanding of Smart Home Behavior, Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, Volume 4, Issue 4, December 2020, Article No.: 124, pp 1–24

[6] Chris Elsden, Tom Feltwell, Shaun W. Lawson, John Vines: Recipes for Programmable Money. CHI 2019: 251

[7] Simone Gallo, Marco Manca, Andrea Mattioli, Fabio Paternò, Carmen Santoro (2021) Comparative Analysis of Composition Paradigms for Personalization Rules in IoT Settings. End-User Development. IS-EUD 2021. Lecture Notes in Computer Science, vol 12724. Springer

[8] Giuseppe Ghiani, Marco Manca, Fabio Paternò, and Carmen Santoro, 2017. Personalization of Context-dependent Applications through Trigger-Action Rules. ACM Transactions on Computer-Human Interaction, 24(2), Article 14.

[9] W He, J Martinez, R Padhi, L Zhang, B Ur, When smart devices are stupid: negative experiences using home smart devices, 2019 IEEE Security and Privacy Workshops (SPW), 150-155

[10] Justin Huang, and Maya Cakmak. 2015. Supporting mental model accuracy in trigger-action programming. In Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '15). ACM, New York, NY, USA, 215-225. DOI=http://dx.doi.org/10.1145/2750858.2805830

[11] Nicola Leonardi, Marco Manca, Fabio Paternò, Carmen Santoro, Trigger-Action Programming for Personalising Humanoid Robot Behaviour, ACM Conference on Human Factors in Computing Systems (CHI'19), Glasgow, Paper 445.

[12] Q. Vera Liao, Daniel M Gruen, Sarah Miller, Questioning the AI: Informing Design Practices for Explainable AI User Experiences, CHI 2020

[13] Marco Manca, Fabio Paternò, Carmen Santoro, Luca Corcella, Supporting end-user debugging of trigger-action rules for IoT applications, International Journal of Human-Computer Studies, Vol.123, 56-69

[14] Marco Manca, Fabio Paternò, Carmen Santoro, Remote Monitoring of End-User Created Automations in Field Trials, Journal of Ambient Intelligence and Humanized Computing, 2021

[15] Andrea Mattioli & Fabio Paternò (2021) Recommendations for creating trigger-action rules in a block-based environment, Behaviour & Information Technology, 40:10, 1024-1034, DOI: 10.1080/0144929X.2021.1900396

[16] Antti Salovaara, Andrea Bellucci, Andrea Vianello, and Giulio Jacucci. 2021. Programmable Smart Home Toolkits Should Better Address Households' Social Needs. In CHI 21, May 8–13, 2021, Yokohama, Japan. ACM, 14 pages.

[17] Ben Shneiderman (2020) Human-Centered Artificial Intelligence: Reliable, Safe & Trustworthy, International Journal of Human–Computer Interaction, 36:6, 495-504

[18] Vijay Srinivasan, Christian Koehler, and Hongxia Jin. 2018. RuleSelector: Selecting Conditional Action Rules from User Behavior Patterns. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 2, 1, Article 35 (March 2018), 34 pages.

[19] Blase Ur, Erin A McManus, Melwyn Pak Yong, Michael L Littman. 2014. Practical trigger-action programming in the smart home. Proceedings of CHI '14. ACM, 803–812. https://doi.org/10.1145/2556288.2557420

[20] Rayoung Yang, Mark Webster Newman, Learning from a learning thermostat: lessons for intelligent systems for the home. 2013. ACM international joint conference on Pervasive and ubiquitous computing, pp. 93-102

[21] L. Zhang, W. He, O. Morkved, V. Zhao, M. L. Littman, S. Lu, and B. Ur. 2020. Trace2TAP: Synthesizing Trigger-Action Programs from Traces of Behavior. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 4, 3, Article 104 (September 2020), 26 pages. https://doi.org/10.1145/3411838