

# Situational-Context: A Unified View of Everything Involved at a Particular Situation

Javier Berrocal<sup>1</sup>(✉), Jose Garcia-Alonso<sup>1</sup>, Carlos Canal<sup>2</sup>, and Juan M. Murillo<sup>1</sup>

<sup>1</sup> University of Extremadura, Cáceres, Spain

{jberolm, jgaralo, juanmamu}@unex.es

<sup>2</sup> University of Málaga, Málaga, Spain

canal@lcc.uma.es

**Abstract.** As the interest in the Web of Things increases, specially for the general population, the barriers to entry for the use of these technologies should decrease. Current applications can be developed to adapt their behaviour to predefined conditions and users preferences, facilitating their use. In the future, Web of Things software should be able to automatically adjust its behaviour to non-predefined preferences or context of its users. In this vision paper we define the Situational-Context as the combination of the virtual profiles of the entities (things or people) that concur at a particular place and time. The computation of the Situational-Context allow us to predict the expected system behaviour and the required interaction between devices to meet the entities' goals, achieving a better adjustment of the system to variable contexts.

**Keywords:** Internet of Things · Web of Things · Context-Aware

## 1 Introduction

The increased capabilities of embedded devices has enabled the development of smart things. These devices may be connected to the Internet, providing a virtual representation of themselves with which other devices can interact, enabling the development of the Internet of Things (IoT) [11]. The Web of Things (WoT) integrates the connected smart things in the web, facilitating their interactions with people [13]. One of the main goals of these paradigms is to simplify people life by making the technology work for them, either providing more information for decision-making or facilitating the accomplishment of some tasks.

It has been predicted that by 2020 there will be 50 to 100 billion of these devices connected to the Internet [23]. However, if we analyse the current state of how people interact with them, the benefits provided will not be as groundbreaking as expected. The reason is that, in order to increase usability, the behaviour of smart things and applications depends on the users preferences and their context, which can shift considerably overtime. However, how systems adapt to these context is still too manual. Manually configuring an increasing number of smart things connected to daily life activities will need too much attention. Moreover,

when there are changes in the context, these devices should be reconfigured. As an example we can consider a thermostat that allows its users to monitor and change their house temperature. To control this system the user establish a set of desired temperatures for specific times. This configuration can be manually overwritten when the user preferences or context change, for example if she is going to arrive home earlier she can put the heating on earlier. This manual control of WoT systems, that is acceptable when working with a small number of devices, will become a burden for users involved in dozens of systems.

Accordingly, solutions are needed to transparently and effortlessly integrate the people's needs, moods and preferences into the connected world of the WoT. There are researches working on gathering and processing the contextual information of users in order to create more comprehensive virtual profiles [1, 10, 15]. Even, the authors of this paper proposed to use the smartphone as a key element to create and maintain these profiles [12]. Nevertheless, an accurate and comprehensive virtual profile is not enough. Techniques to adapt the software behaviour to the context are also necessary. Currently, researchers are working on techniques such as Dynamic Composition [6] or Context Oriented Programming (COP) [14]. These techniques allow developers to predefine different behaviour of an applications depending on the identification of specific contextual information. They concur that the information and/or variables triggering the adaptations is detailed within the source code of the applications. Thus, the adaptation capabilities are limited to the set of predefined contexts. However, the variability of this information is very large and is difficult to identify and express every plausible situation in the development phase, especially for everyday environments in which interactions depend on each user and her context, i.e. her preferences, the people located around her, their history, etc. For example, when a person get into her home, the smartphone may automatically interact with the thermostat to establish the desired temperature, but this interaction can also depend on whether she is alone or accompanied, her mood, etc.

Here, we present a vision paper in which the authors outline a set of concepts for achieving adaptation in the defined context. Concretely, we propose the new concept of Situational-Context as a way to analyse the conditions that exist at a particular time and place; and how this analysis can be used to predict, at run-time, the expected behaviour of WoT systems. The Situational-Context is defined as the resulting context of composing the virtual profiles of the different entities (things and people) involved in a particular situation. In this composed context there will be entities providing goals that details specific conditions that are desirable to be achieved, and entities providing skills enabling the fulfilment of these goals. Thus, once the Situational-Context is composed, the ways in which the entities will better satisfy the goals should emerge from the Situational-Context itself. The Situational-Context will provide a higher level of automation of smart things with people. Currently, there is a large amount of works related with the Situational-Context in the Context-Aware [15], Ubicomp [5], User Modelling [16] and Ambient Intelligence [20] areas. Some of their results will be used to develop its technical aspects. The contribution of the Situational-Context to WoT will be not only the unified view of the virtual profiles and the emerging

interactions from the situations, but also new programming models to develop applications aware of the situations will have to be defined.

To define the Situational-Context the rest of the paper is structured as follows. Section 2 presents the motivations and some related work. Section 3 details the Situational-Context concept, the technology required to support it and a proof of concept. And Sect. 4 presents some conclusions and future works.

## 2 Motivations and Related Works

In the WoT usually several devices are orchestrated to build complex systems [17]. However, as the WoT is more integrated into people daily activities this orchestration becomes more complex. The Ambient Intelligence (AmI) has emerge as a disciple for making the everyday environments sensitive and responsive to people [20]. AmI needs to be aware of the users preferences in order to know when a device should acts. This is even more challenging when the needs of a multiple entities should be analysed in order to predict the action to perform [7].

In this sense, there are different researchers focused on the identification of people's context. Concretely, the authors of this paper have been working on the People as a Service (PeaaS) and the Internet of People (IoP) approaches. PeaaS [12] is a mobile-centric computing model to infer the context of smartphones' owners and generate their sociological profile. IoP [21] propose an infrastructure and a manifesto for WoT systems that support this proactive adaptations. This manifesto indicates that the interactions between things and people must *be social*, must *be personalized* with the users profiles, must *be predictable*, and must *be proactive* and automatically triggered depending on the context.

The raw contextual information related to a person or a thing is very rich, however some times it can be too basic. Other research works focus on computing the raw information for making high level context deductions. In [22] the authors propose a system that can automatically recognize the high-level context of the users, i.e. activities, emotions, and relationships with other users. In [10], the authors indicate that the user context can be expressed as a combination of the user's activity, light conditions, social setting and geographical location. So, they propose a system to gather the user context and perform high level inferences.

One of the main goals of inferring high level information is to better adapt the applications behaviour to the users. The same authors of [10] reuse the deducted information to adapt the interface of an app to the user environment. The COP paradigm provides an additional dimension to standard programming techniques to dynamically switch among the behaviours associated with each context [24]. Most of the approaches defined in this paradigm group behaviour in layers related with a specific context. The activation of a layer is usually predetermined at the development stage. Even there are works decoupling the context from the layers, providing greater flexibility [19]. The Dynamic Composition paradigm is a step forward when the interactions between devices cannot be identified at the development stages. It allows developers to implement the application behaviour

without defining the specific devices involved. Therefore, applications choose the devices involved in a specific interaction at run-time.

Therefore, there are a lot of proposals for building comprehensive virtual profiles. However, the techniques for developing systems adaptable to the users' profiles requires to predefine in the development phase when each behaviour is activated. This limits the customization of applications and makes it difficult to obtain WoT systems totally responsive to users. It would be desirable that the behaviours and interactions emerge from the concrete situations and that the system would be able to respond in an ad-hoc way to each situation.

The next section focuses on detailing how the context of the devices can be computed to identify situations of people and things, and how such situations can be used to predict at run-time the interactions to trigger.

### 3 Emerging Interactions from the Situational-Context

#### 3.1 Situational-Context

The Situational-Context can be defined as the composition of the virtual profiles of all the entities involved in a situation. For a meaningful composition of these profiles, we consider that they contain, at least, the following information:

- A *Basic Profile* containing the dated raw information with the entity's status, the relationships with other devices and its history. This profile can be seen as a timeline with the changes and interactions that happened to the entity.
- *Social Profile*. This profile contains the results of high level inferences performed over the Basic Profile.
- The *Goals* detailing the status of the environment desired by the entity. These Goals can also be deduced from the Basic and Social Profiles.
- The *Skills* or capabilities that an entity has to make decisions and perform actions capable of modifying the environment and aimed at achieving Goals.

The result of composing the virtual profiles of the involved entities is not only the combined information of all entities. It contains the combined history of the entities ordered in a single timeline, the result of high level inferences performed over the combined virtual profiles, the set of Goals of the entities and their Skills. From the combined information of the Situational-Context, strategies to achieve Goals based on the present Skills should be identified. These strategies will guide the prediction of the interactions that must emerge from the context.

Furthermore, the Situational-Context is a dynamic abstraction of the combined profiles and therefore evolves through time. To analyse the instantaneity of this context, we use the concept of *Configuration*. A Configuration is the unified and stable view of the virtual profiles of the devices involved in the situation at a specific point in time. When changes in the environment happen, the Configuration is no longer stable and must be updated. Thus, a new Configuration must be defined from the updated/new virtual profiles of the devices. Thus, the Situational-Context can also be seen as a succession of Configurations.

Figure 1 shows the Situational-Context for controlling the temperature of a room. It contains a first configuration (C1) combining the virtual profiles of a thermostat and the smartphone of a person that is in the room. The smartphone defines the Goal to have a comfort temperature and the thermostat has a Skill to control the temperature of the room. When a new user with the same Goal in her profile enters the room, the situation change, a new configuration (C2) is computed and the strategies required to achieve the combined Goals are identified. Then, the interactions required for setting the adequate comfort temperature will emerge from this context.

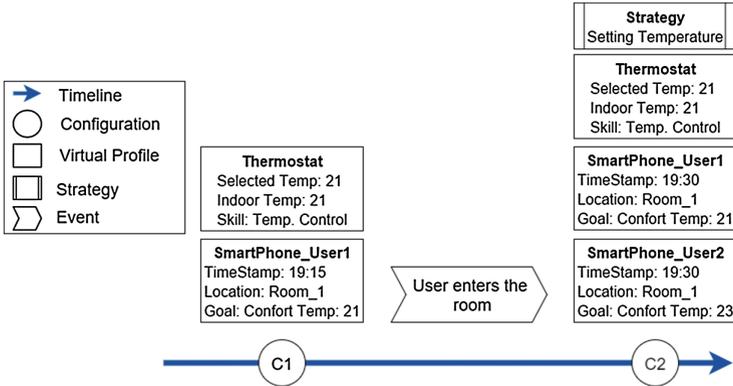


Fig. 1. Excerpt of a Situational-Context.

### 3.2 Technology for Supporting the Situational-Context

To support the Situational-Context, as described above, there are a number of technological issues that must be resolved.

First, which device or devices should compute the Situational-Context? This computation can be done in a Cloud environment. However, in order to reduce the network overhead and thanks to the increased capabilities of smart things, this computation can also be done either by a local device or distributively by a set of local devices [2]. Currently, Multi-Agent Systems [3] can be used to develop self-organised, reconfigurable and proactive systems. It should be evaluated what technique is the most appropriate for computing the Situational-Context or even if a combination of different techniques should be applied depending on the size and type of the virtual profiles to combine.

Second, how contextual information can be exchanged? There is a wide range of technologies for this purpose. There are middlewares homogenizing the communications in heterogeneous networks [8]. And there are works, like the SOFIA project [25], creating a semantic interoperability platform for making the information available for smart services. It is necessary to identify whether they can be used for identifying the devices involved in a configuration and to manage the interactions between them.

Third, how common Goals should be agreed for a specific configuration when the devices involved have different or, even, opposed Goals? Currently, there are algorithms for negotiating which device should perform a command [4] when in the surrounding there are several that have the capabilities to respond to it. In the Situational-Context, negotiation algorithms are needed to autonomously agree on a common Goal.

Fourth, how the strategies to achieve Goals should be identified? They can be predefined in order to be triggered depending on the Goals to achieve, but again this would compromise the applications flexibility. Should they also emerge from each specific configuration? The Spatial and Temporal reasoning areas [9] have been previously used to get a better understanding of the context in order to make sensible decisions. It should be evaluated whether is possible to infer the strategy to execute from the context using these techniques.

Finally, how the interactions can emerge from the Situational-Context and from each Configuration? The Self-Adaptive [18] software systems can modify themselves at run-time and, as detailed above, there are different proposals to develop applications that adapt their behaviour to the context. It should be assessed whether these approaches can be used to develop applications that autonomously compose the devices involved in a Configuration and trigger specific actions depending on the established Goals and Strategies.

### 3.3 Proof of Concept

Currently, we are working on a proof of concept for the computation of the Situational-Context. This proof of concept consist in an Android set-top box connected to a temperature sensor and an air conditioner, and a mobile app to control the temperature of a room that is installed on the users' phones.

The Android device has an application for controlling the air conditioner, for getting the room temperature from the sensor and for broadcasting information about the device. The mobile app sends commands for getting the temperature, stores the user contextual information and has rules for high level deductions and for its definition as Goals. Specifically, it has a rule to infer the user's comfort temperature from manual interactions.

In this case, the Situational-Context is locally computed by the smartphone. To that end, it constantly monitor the surrounding devices broadcasting information, identifies which ones have Skills for controlling the temperature, queries them to obtain the room temperature and, if it is different than the comfort temperature, the mobile app proactively trigger an strategy to set a new one.

Finally, this strategy has also defined a negotiation algorithm to agree a common temperature when there are several users. This algorithm, first, identifies all the devices involved in the conflict. For this, it broadcast a signal indicating that it wants to change the temperature, this signal is replied by the devices with a Goal defined involving the temperature. Subsequently, a communication is established between them to exchange the comfort temperature. The final temperature is the average of all temperatures and is notified to the Android device by the device that started the algorithm.

The tests conducted so far show us that the computation of the Situational-Context is feasible. However, more research work is still needed for the computation of complex situations.

## 4 Conclusions and Future Work

Current WoT applications can be implemented to have a specific behaviour depending on the preferences and the context of users. However, this adaptation is limited to the behaviours defined in the development phase. In the future these applications shall be fully self-adaptive and able to completely change their behaviour, at run-time, to cover the needs of any user or any group of users, and to use the capabilities of the new devices included in the system.

Here, we present a vision paper basing this adaptation in the Situational-Context. This facilitates the identification of the goals that should be pursued by the surrounding devices and the needed strategies and interactions to achieve them. We are currently working on formalizing the Situational-Context and its computation. For computing the Situational-Context, new programming models for the different development phases of WoT applications will be needed. These models will facilitate the development systems aware of the environment and able to react to its needs.

**Acknowledgments.** This work was supported by the Spanish Ministry of Science and Innovation (TIN2014-53986-REDT, TIN2015-67083-R and TIN2015-69957-R), by the Department of Economy and Infrastructure of the Government of Extremadura (GR15098), and by the European Regional Development Fund.

## References

1. Abowd, G.D., Dey, A.K.: Towards a better understanding of context and context-awareness. In: Gellersen, H.-W. (ed.) HUC 1999. LNCS, vol. 1707, pp. 304–307. Springer, Heidelberg (1999)
2. Arslan, M., Singh, I., Singh, S., Madhyastha, H., Sundaresan, K., Krishnamurthy, S.: CWC: a distributed computing infrastructure using smartphones. *IEEE Trans. Mob. Comput.* **14**(8), 1587–1600 (2015)
3. Barbati, M., Bruno, G., Genovese, A.: Applications of agent-based models for optimization problems: a literature review. *Expert Syst. Appl.* **39**(5), 6020–6028 (2012)
4. Bauerle, F., Miller, G., Nassar, N., Nassar, T., Penney, I.: Context sensitive smart device command recognition and negotiation. In: Giaffreda, R., Vieriu, R.-L., Pasher, E., Bendersky, G., Jara, A.J., Rodrigues, J.J.P.C., Dekel, E., Mandler, B. (eds.) *IoT360 2014*. LNICST, vol. 150, pp. 314–330. Springer, Heidelberg (2015)
5. Caceres, R., Friday, A.: Ubicomp systems at 20: progress, opportunities, and challenges. *IEEE Pervasive Comput.* **1**, 14–21 (2011)
6. Chen, G., Li, M., Kotz, D.: Data-centric middleware for context-aware pervasive computing. *Pervasive Mob. Comput.* **4**(2), 216–253 (2008)
7. Cook, D.J., Augusto, J.C., Jakkula, V.R.: Ambient intelligence: technologies, applications, and opportunities. *Pervasive Mob. Comput.* **5**(4), 277–298 (2009)

8. Dubois, D., Bando, Y., Watanabe, K., Miyamoto, A., Sato, M., Papper, W., Bove, V.: Supporting heterogeneous networks and pervasive storage in mobile content-sharing middleware. In: Consumer Communications and Networking Conference, pp. 841–847 (2015)
9. Galton, A.: *Qualitative Spatial Change*. Oxford University Press, Oxford (2000)
10. Gronli, T.M., Ghinea, G., Younas, M.: Context-aware and automatic configuration of mobile devices in cloud-enabled ubiquitous computing. *Personal Ubiquit. Comput.* **18**(4), 883–894 (2014)
11. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of things (IoT): a vision, architectural elements, and future directions. *Future Gener. Comput. Syst.* **29**(7), 1645–1660 (2013)
12. Guillen, J., Miranda, J., Berrocal, J., Garcia-Alonso, J., Murillo, J.M., Canal, C.: People as a service: a mobile-centric model for providing collective sociological profiles. *IEEE Softw.* **31**(2), 48–53 (2014)
13. Guinard, D., Trifa, V., Mattern, F., Wilde, E.: From the internet of things to the web of things: resource-oriented architecture and best practices. In: Uckelmann, D., Harrison, M., Michahelles, F. (eds.) *Architecting the Internet of Things*, pp. 97–129. Springer, Heidelberg (2011)
14. Hirschfeld, R., Costanza, P., Nierstrasz, O.: Context-oriented programming. *J. Object Technol.* **7**(3), 125–151 (2008). ETH Zurich
15. Hong, J.Y., Suh, E.H., Kim, S.J.: Context-aware systems: a literature review and classification. *Exp. Syst. Appl.* **36**(4), 8509–8522 (2009)
16. Kobsa, A.: Generic user modeling systems. *User Model. User-Adap. Interact.* **11**(1–2), 49–63 (2001)
17. Kovatsch, M.: CoAP for the web of things: from tiny resource-constrained devices to the web browser. In: *ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication*, pp. 1495–1504. ACM, New York (2013)
18. Macías-Escrivá, F.D., Haber, R.E., Toro, R.M., Hernández, V.: Self-adaptive systems: a survey of current approaches, research challenges and applications. *Expert Syst. Appl.* **40**(18), 7267–7279 (2013)
19. Maingret, B., Mouël, F.L., Ponge, J., Stouls, N., Cao, J., Loiseau, Y.: Towards a decoupled context-oriented programming language for the internet of things. In: *International Workshop on Context-Oriented Programming*, pp. 1–6. ACM (2015)
20. Marzano, S.: *The New Everyday: Views on Ambient Intelligence*. 010 Publishers, Rotterdam (2003)
21. Miranda, J., Makitalo, N., Garcia-Alonso, J., Berrocal, J., Mikkonen, T., Canal, C., Murillo, J.: From the internet of things to the internet of people. *IEEE Internet Comput.* **19**(2), 40–47 (2015)
22. Park, H.S., Oh, K., Cho, S.B.: Bayesian network-based high-level context recognition for mobile context sharing in cyber-physical system. *Int. J. Distrib. Sens. Netw.* **2011**, 10 (2011)
23. Perera, C., Liu, C.H., Jayawardena, S., Chen, M.: Context-aware computing in the internet of things: a survey on internet of things from industrial market perspective. *CoRR* (2015)
24. Salvaneschi, G., Ghezzi, C., Pradella, M.: Context-oriented programming: a software engineering perspective. *J. Syst. Softw.* **85**(8), 1801–1817 (2012)
25. SOFIA Project: Smart Objects For Intelligent Applications (2009). <http://www.sofia-project.eu/>