

A Cognitive Management Framework for Empowering the Internet of Things

Vassilis Foteinos¹, Dimitris Kelaidonis¹, George Poulios¹, Vera Stavroulaki¹,
Panagiotis Vlacheas¹, Panagiotis Demestichas¹, Raffaele Giaffreda²,
Abdur Rahim Biswas², Stephane Menoret³, Gerard Nguengang³, Matti Etelapera⁴,
Nechifor Septimiu-Cosmin⁵, Marc Roelands⁶,
Filippo Visintainer⁷, and Klaus Moessner⁸

¹ Department of Digital Systems, University of Piraeus, Greece
{vfotein,dkelaid,gpoulios,veras,panvlah,pdemest}@unipi.gr

² CreateNet, Italy

{raffaele.giaffreda,abdur.rahim}@create-net.org

³ Thales Communications, France

{stephane.menoret,gerard.nguengang}@thalesgroup.com

⁴ VTT Research Centre, Finland

Matti.Etelapera@vtt.fi

⁵ Siemens, Romania

septimiu.nechifor@siemens.com

⁶ Alcatel Lucent Bell Labs, Belgium

marc.roelands@alcatel-lucent.com

⁷ Centro Ricerche FIAT

filippo.visintainer@crf.it

⁸ University of Surrey, Guildford, UK

K.Moessner@surrey.ac.uk

Abstract. This work presents a Cognitive Management framework for empowering the Internet of Things (IoT). This framework has the ability to dynamically adapt its behaviour, through self-management functionality, taking into account information and knowledge (obtained through machine learning) on the situation (e.g., internal status and status of environment), as well as policies (designating objectives, constraints, rules, etc.). Cognitive technologies constitute a unique and efficient approach for addressing the technological heterogeneity of the IoT and obtaining situation awareness, reliability and efficiency. The paper also presents a first indicative implementation of the proposed framework, comprising real sensors and actuators. The preliminary results of this work demonstrate high potential towards self-reconfigurable IoT.

Keywords: Cognitive Management, Composite Virtual Objects, Internet of Things, Stakeholder requirements, User requirements, Virtual Objects.

1 Introduction

The “7 trillion devices for 7 billion people” paradigm as described in [1] yields that the handling of the amount of objects that will be part of the Internet of Things (IoT)

requires suitable architecture and technological foundations. Internet-connected sensors, actuators and other types of smart devices and objects need a suitable communication infrastructure. At the same time, the lack in terms of management functionality overcome the technological heterogeneity and complexity of the pervasive networks, calls for the definition of mechanisms for enhanced situational awareness. Such mechanisms should provide high reliability through the ability to use heterogeneous objects in a complementary manner for reliable service provision. Moreover energy-efficiency should be enabled through the selection of the most efficient and suitable objects from the set of heterogeneous ones. The sheer numbers of objects and devices that have to be handled and the variety of networking and communication technologies, as well as administrative boundaries that have to be supported require a different management approach. The idea is to enable seamless and interoperable connectivity amongst heterogeneous number of devices and systems, hide their complexity to the users/stakeholders while providing sophisticated services [2].

This work presents a Cognitive Management framework targeted to overcoming this technological heterogeneity and complexity. This framework comprises capabilities to dynamically select its behaviour (managed system's configuration), through self-management functionality, taking into account information and knowledge (obtained through machine learning) on the context of operation (e.g., internal status and status of environment), as well as policies (designating objectives, constraints, rules, etc.). The framework comprises three main levels of enablers, namely the *Virtual Object (VO)*, *Composite Virtual Object (CVO)* and *User/Stakeholder and Service* levels, which are reusable for the realization of diverse applications such as ambient assisted living, smart office (easy meeting), smart transportation, supply chain management [3]. In each layer there are scalable fabrics, which offer mechanisms for the registration, look-up and discovery of entities and the composition of services. Cognitive entities at all levels provide the means for self-management (configuration, healing, optimization, protection) and learning. In this respect, they are capable of perceiving and reasoning on their context/situation (e.g., based on event filtering, pattern recognition, machine learning), and of conducting associated knowledge-based decision-making (through associated optimization algorithms and machine learning).

Virtual objects (VOs) are primarily targeted to the abstraction of technological heterogeneity. VOs are virtual representations of real world objects. (e.g., sensors, actuators, devices, etc.). A CVO is a cognitive mash-up of semantically interoperable VOs that renders services in accordance with the user/stakeholder perspectives and the application requirements. User/stakeholder related objects convey the respective requirements. Such entities are capable of detecting users/group of users' intentions and behaviour, inferring, and eventually acting on behalf of the users. In this respect, there is seamless support to users, which is in full alignment with their requirements (the learning capabilities of the cognitive entities of this layer are applied for acquiring knowledge on user/stakeholder preferences, etc.). Furthermore, this framework comprises three main operations, the *Dynamic Creation of a CVO*, the *Knowledge-based Instantiation of a CVO* and the *Self-healing of a CVO*. The first operation corresponds to the creation of a CVO from scratch, in order to provide the requested service to the users. The second operation enables the reuse of an already existing CVO.

The self-healing operation is responsible for finding the optimal alternative object, when a Real World Object (RWO) is unreachable and through its VO to sustain the CVO's functionality.

The rest of the paper is structured as follows. Section 2 provides an overview of related work regarding the concepts of VOS, CVOs and management frameworks for IoT. Section 3 provides a description of the Cognitive Management framework for IoT, elaborates on the definition of VOs and the mechanisms of the VO level, the CVO management mechanisms and discusses means provided by the User/Stakeholder and Service level. Section 4 focuses on a first implementation of the proposed framework presenting some of the operations that it enables and presents an indicative application example in an ambient assisted living scenario. Finally, a summary of the key points addressed in this work is provided in section 5.

2 Related Work and Paper Contribution

The concept of virtual representations of devices/objects in the scope of the IoT is a key issue in most of the Future Internet, IoT projects and research initiatives. Authors in [3] introduce the concept of the Versatile Digital Item (VDI), which is a package of information about services, people and physical objects, independently of the structure or geographical location of the context. Work in [5] aims at creating an open, business driven architecture that addresses the scalability problems for a large number of globally distributed wireless sensor and actuator networks [6]. Additionally, a platform, which among others aims to transform every device into a web service with semantic resolution, is described in [7]. The platform features a Service Oriented Architecture based on open protocols and middleware, with the aim of transforming every subsystem or device into a web service with semantic resolution.

Virtual representations of Real World Objects (RWOs) are also addressed in [8] with the target to make physical world "information" available for smart services - connecting the physical world with the information world. Smart-M3 [9], a core component of Open Innovation Platform, aims at providing the baseline for a solution to cross-domain and cross-platform interoperability platform and information exchange. Smart-M3 makes it possible to mash up and integrate information among all applications and domains ranging from different embedded domains to the Web.

The concept of the "ucode", an identification number assigned to tangible "physical objects" and "places" is one of the work topics of the Japanese Ubiquitous ID Center [10]. This concept of "ucodes" can also be applied for content and information which do not exist in the real world and for more abstract concepts. One of the most recent works relevant to virtual representations of devices/objects is the concept of the "Web of Things" discussed in [11], where the aim is to integrate RWOs, data and functionality into the Web, instead of exposing these through vertical system designs. More specifically it is suggested that the exploitation of Web technologies is a suitable solution for building applications on top of services offered by smart things.

Regarding research on VOs and CVOs semantics, various activities also address the issue as a means for interoperability. An example work is [12] which aims to

investigate unified concepts, methods, and software infrastructures that facilitate the efficient development of applications that span and integrate the Internet and the embedded world.

The framework presented in this work targets a variety of application domains and advanced business processes, while being agnostic of the application domain in which it will be applied, enabling cross-domain re-use of technology and exchange of information. The VO concept is introduced as a dynamic virtual representation of real world/digital objects. Furthermore, the concept of the CVO is introduced, as a collection of information and services from partial digital images of the world and their VOs. The CVO concept leads to intelligent services, fulfilling requirements (also from hidden stakeholders), while details and complexity are hidden from end users. The proposed framework comprises cognitive mechanisms to enable the dynamic creation, invocation, deployment, self-configuration, self-management of VOs, CVOs and services.

3 Cognitive Management Framework for IoT

The Cognitive Management framework for IoT consists of the VO layer, the CVO layer and the User/Stakeholders and Service layers, as already introduced. Figure 1 depicts the aforementioned layers of the proposed Cognitive Management framework. This framework, enables the handling of diverse objects and the functions and services these objects provide. Thus, it will enable the realization of a wider IoT ecosystem which can be exploited by many different types of users and stakeholders (across different application and usage domains). While RWOs may be owned (controlled) by a particular stakeholder, VOs (i.e. the abstractions of real world or digital objects) can be owned (controlled) by particular service providers. In turn, CVOs may be owned (controlled) by yet another provider who adds value by combining different VOs and providing these combinations to users. This hierarchical structure leads to a rather complex eco-system, that opens new opportunities for various stakeholders.

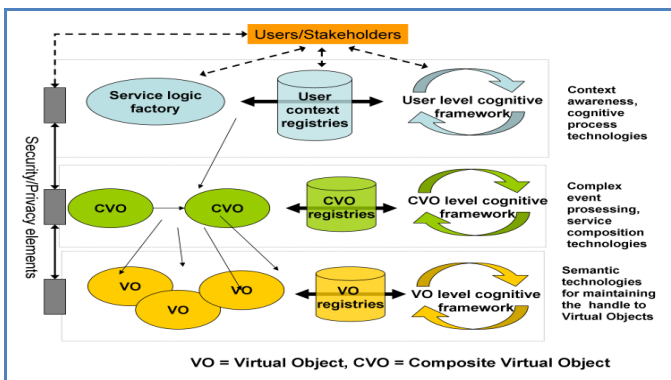


Fig. 1. Layers of the Cognitive Management framework [3]

Furthermore, the Cognitive Management framework will ensure that the complexity of this eco-system will be well hidden from the different players and stakeholders.

3.1 Virtual Objects Level

The world of IoT is characterised by two main features: (i) considerable number of objects that fall within the scope of any IoT application and (ii) inherently unreliable nature of such objects. The purpose of VOs is to efficiently tackle such issues and hide the complexity of the IoT infrastructure. VOs are virtual representations of RWOs and/or digital objects that can enhance IoT, whereby real objects become virtually “always-on”. A VO can be dynamically created and destructed, may consist of information, services and is a dynamic object since it has to represent dynamically changing RWOs. Cognitive mechanisms at the VO level enable self-management and self-configuration of RWOs. Moreover, the introduction of cognitive mechanisms will lead to knowledge on how RWOs react to specific situations. Thus, the operation and control of these objects will become more efficient.

The VO level comprises mechanisms that provide awareness about the physical objects presence and relevance. A complete knowledge about physical objects will be necessary to maintain the association between the VOs and the real objects they represent. This knowledge is used to have a constant handle (the VO abstraction) on the best suited physical objects to keep association with, regardless of their mobility or failure of some of the links to them etc. In other words, the VO level comprises mechanisms for monitoring the status/capabilities of physical objects and controlling the various links with physical objects to make sure that the VOs are resilient, even when the associated physical objects might be temporarily unavailable. Another important aspect of this level is optimization from the point of view of resources. In this direction, energy plays an important role in the future IoT applications. Physical objects have varied energy supplies – some of them are battery operated and some are mains/grid powered. It makes sense that VOs representing IoT devices that have high energy supply could take over the tasks rather than VOs that have less supply. In fact, while forming a CVO, VOs need to be selected carefully to get better overall resource (mainly energy but other resources are also considered) profiling. However, while optimizing the resources, dependability of VOs and thus CVOs should also be taken care of. For example, more VOs (in turn more number of hops) could reduce the reliability of a CVO. This is critical for some applications such as health, which require higher reliability.

3.2 Composite Virtual Objects Level

A CVO is a cognitive mash-up of semantically interoperable VOs that renders services in accordance with the user/ stakeholder perspectives and the application/service requirements. Cognitive mechanisms are exploited to enable reuse of existing VOs and CVOs by various applications, potentially also outside the context for which they were originally developed. In this sense, the CVO level encompasses functionalities that allow finding the optimal possible way to deliver the application/service, given:

(i) the requirements coming from the user level; (ii) the capabilities offered by VOs and (iii) existing CVOs' functionality. A CVO is in charge of rendering the requested service, in a manner that is most appropriate with respect to the profile and situation of the user. It is one of the CVO responsibilities to guarantee the availability, performance and dependability of the application, and therefore, of the constituent services. In this respect, the CVO will be in charge of selecting the best VOs (and potentially other CVOs), orchestrating their operation, handling faults, and eventually, releasing the VOs. The selection will be based on the appropriateness of the VO/CVO, e.g., in terms of relevance to the application, situation and profile information and knowledge, and on their status, e.g., in terms of computing, storage, transmission and energy resources. Each CVO will also possess its own (local) cognition capabilities. These derive from learning about the VO existence, behaviour and capabilities as well as monitoring and learning situation aspects that are relevant to the CVO operation. This will be used for optimizing the CVO functionality and the management of the comprised VOs during the CVO life-cycle.



Fig. 2. Request and Situation Matching mechanism

DECISION MAKING

PAUSE RUNNING(1) - SENDING CVO 1 SET STEP

REQUEST FOR NEW CVO:

Emergency Alarm Body Temperature Room Humidity Lamp Room Luminosity Body Pulses Room Heating Room Temperature
 [Qoa: 0.32 , Per: 0.32 , Sec: 0.32 , Exp: 0.01 , Net: 0.01 , Ene: 0.01]

AVAILABLE VOs

ID	FUNCTION	LOCATION	QUA	PER	SEC	EXP	NET	ENE	AG-VAL
http://icorevo/VO111	Car Alarm	562 98 - 120 13			High	High			1
http://www.lot-core.eu/vo/VO115	Body Temperature	395 5- 3613 01			Medium			Low	0.66
http://www.lot-core.eu/vo/VO115	Body Pulses	395 5- 561 01	Low		Medium		Low	Medium	1
http://icorevo/VO108	Fire Alarm	863 - 1015	Low		Medium		Medium	Low	0.99
http://icorevo/VO108	Fire Alarm	863 - 1015	High	High	High	Low	Low	Low	2.94
http://www.lot-core.eu/vo/VO001	Room Temperature	491 05- 407 03	Medium	Medium	Medium	Medium	Medium	Low	2
http://icorevo/VO114	Oven Temperature	846 31 - 437 1			Medium			Low	0.66
http://icorevo/VO101	Room Temperature	314 8- 736 32	Low	Low	Low	High	High	High	1.68
http://www.lot-core.eu/vo/VO201	Room Humidity	740 37- 754 83	High	High	High	Low	Low	Low	2.94
http://www.lot-core.eu/VO202	Room Humidity	836 64 - 830 00	Medium		Medium		Medium	Low	4.52

PROPOSED CVO

ID	FUNCTION
http://icorevo/VO110	Fire Alarm
http://www.lot-core.eu/vo/VO115	Body Temperature
http://www.lot-core.eu/vo/VO201	Room Humidity
http://icorevo/VO006	Lamp
http://www.lot-core.eu/vo/VO202	Room Luminosity
http://www.lot-core.eu/vo/VO115	Body Pulses
http://icorevo/VO004	Room Heating

QOA = QUALITY
 PER = PERFORMANCE
 SEC = SECURITY
 EXP = EXPENDITURES
 NET = NETWORK
 ENE = ENERGY
 AG-VAL = AGGREGATE VALUE

Solution Value: 20.43

Telecommunication Networks & Integrated Services Laboratory University of Piraeus

Fig. 3. Decision Making mechanism

Two core mechanisms of the CVO level addressed here are the *Request and Situation Matching* (Figure 2) and the *Decision Making* (Figure 3) [13]. The goal of the *Request and Situation Matching* mechanism is to identify past service requests that

match closely enough to the present ones and the situations under which they were issued, so that the task of VO composition from scratch can be avoided under certain circumstances. In order to compare between past and present situations and requests, parameters that describe them have been identified. Request parameters consist of the set of functions and policies requested. Situation parameters consist of the time of day the request occurred, the area of interest, and the available VOs at that time. In order to enhance the filtering process, the requested functions can be matched with approximate ones, e.g. a video capture function can satisfy a requirement for an image capture function. More details on the matching process can be found in [13]. The result of this mechanism is performance gain (mainly in terms of time), as the process of creating a CVO from scratch is more complex. Moreover, reusability of resources is introduced in the system. The *Decision Making* mechanism is triggered by the Request and Situation Matching mechanism. Its objective is to find the optimal composition of VOs that fulfils the requested functions and policies. It receives as input a set of available VOs, a set of functions for each VO, a set of features for these VOs and finally the requested policies. The result of the Decision Making process is the creation and activation of a new CVO. The description of the newly instantiated CVO is recorded in the CVO registry in order to be available for future requests.

3.3 User/Stakeholder and Service Level

From the users/applications perspective, three concepts, i.e. IoT, ubiquitous computing, and ambient intelligence aim at delivering smart services to users [14]. A part of the smartness relies on situation awareness, e.g., service provision according to the needs that exist at the place, time and overall situation. Also, at a societal level, smartness also requires that the needs of diverse users and stakeholders are taken into account. Stakeholders can be the owners of the objects and of the communication means. Different stakeholders that are part of the current Internet environment and will be part of the Future Internet, have interests that may be adverse to each other and their very own criteria on how objects could be used and should be accessed. So, a key challenge that needs to be tackled includes the handling of the diversity of information while respecting the business integrity, the needs and rights of users and of the various stakeholders. At current, the interactions between users/stakeholders and RWOs or “things”, are mainly manual. Using these RWOs requires normally manual intervention of the user. The user will need information about how to use a “thing”, and how things can be combined for use. This level provides the actual user/service level mechanisms that reside on top of the (C)VOS, and provide the means to leverage the digital representations of RWOs to make IoT applications more cognitive and intention-aware. The User/Stakeholder and Service level comprises mechanisms for the identification of requirements stemming from different types of users, the dependencies they cause, by themselves and across large groups of users, as well as across space and time. Users’ and device situation changes, individually or collectively across the user population, time and space, often exhibit certain patterns, and user behaviour and part of the users’ intentions could potentially be derived from this.

Moreover, this level comprises mechanisms for the appropriate (automated) analysis of application/service requirements for generating the appropriate service logic and service logic-enabling descriptions for instantiating applications and application-enabling functions in the CVO level.

4 Framework Implementation

4.1 Operation

A prototype of the proposed Cognitive Management framework has been implemented. The current corresponding implementation comprises, apart from software components for the various functional components presented in the previous, a number of actual sensors and actuators. This sub-section provides a high-level view of the implemented framework and presents three main operations this enables, namely dynamic CVO creation, knowledge-based CVO instantiation and self-healing of a CVO.

Dynamic Creation of a CVO. A user requests a service and declares the importance of service/function features, denoted as policies, which should be respected by the system. This request is captured through a User Interface at the Service level and is elaborated, in order to extract the requested functions and defined policies. This information is enriched with the situation parameters (area of interest, time of request) and is forwarded to the Request and Situation Matching mechanism. Moreover, learning mechanisms provide information regarding the user preferences. The Request and Situation Matching mechanism searches for an already existing CVO that can fulfill the request, in the CVO registry (Figure 4). If this reuse of a CVO is not possible, then it forwards the relevant information to the Decision Making, in order to create an optimal composition of VOs, according to requested functions and policies. For this purpose, information on available VO features and provided functions (VO descriptions) are retrieved from the VO registry. Once a CVO is created, the request, associated situation and CVO composition are recorded in the CVO registry, in order to ensure that if a similar contextual situation occurs again, the solution may be retrieved directly. Finally, the created CVO provides the requested service to the user (Figure 5).

CVO REGISTRY									
[R] ID	[S] ID	[R] Functions	[R] Policies	[S] Time	[S] Area	[S] Av...	SVO ID	CVO Components	
http://app...	http://apps...	video playback	network.u.i	Thu May 17 10:11:...	x:247.07...	http://...	http://co...	Room Temperature	http://cor...
http://app...	http://apps...	Video Playback	Performance:0.01	Thu May 17 16:17:...	x:400.56...	http://...	http://co...	Room Temperature	http://cor...
http://app...	http://apps...	Security Alarm	Security:0.01	Thu May 17 17:06:...	x:423.32...	http://...	http://co...	Fire Alarm	http://core/vo/...
http://app...	http://apps...	Room Cooling	Quality:0.01	Thu May 17 18:41:...	x:458.54...	http://...	http://co...	Room Cooling	http://core/vo/...
http://app...	http://apps...	Audio Playback	Energy:0.01	Thu May 17 18:43:...	x:458.54...	http://...	http://co...	Audio Playback	http://core/v...
http://app...	http://apps...	Room Cooling	Energy:0.01	Thu May 17 19:26:...	x:458.54...	http://...	http://co...	Room Temperature	http://cor...

Request parameters
(functions, policies)

Situation parameters
(time and geographical area of the request, available VOs)

CVO
composition

Fig. 4. CVO Registry

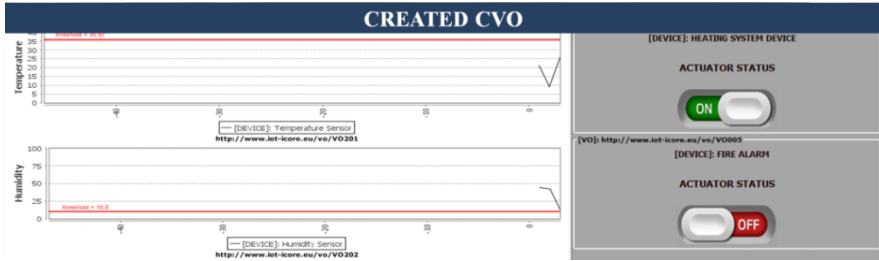


Fig. 5. Created CVO

Knowledge-based Instantiation of a CVO. This process enables the reuse of an already existing CVO. The steps of the previous operation are repeated until the request is received by the Request and Situation Matching component. At this point the service request and situation information can be compared with records in the CVO registry for an adequate match. Past records corresponding to CVO components (VOs) with functions that are unavailable in the current situation (either exact or approximate ones) are filtered out, as they definitely cannot fulfil the service goals. The remaining records are ranked based on a satisfaction-rate similarity metric and the highest ranked one is tested against the similarity threshold. The satisfaction-rate depends on the amount of total requested functions being available as well as their correlations and it is calculated as a score (i.e. sum) of these correlations between the set of the requested and the required CVO functions. Apart from the functions, for the calculation of the overall similarity metric also the rest of the situation and request parameters are taken into account. If the overall similarity metric for an existing CVO equals or exceeds a specific threshold, then this existing CVO can be considered as suitable for a newly issued request. In this way, the CVO level components can apply known solutions as a response to a service request, thus, reducing the time needed for handling of requests from the Service level.

Self-healing of a CVO. The self-healing process enables the detection of a problem in the operation of a used RWO and corresponding VO and dynamic reconfiguration of the corresponding CVO to overcome the problem. This process is triggered when a VO failure is identified due to an RWO becoming unreachable (e.g. due to loss of connectivity of the VO with RWO, RWO hardware failures, etc). A Reconfiguration request is issued from the Request and Situation Matching to the Decision Making component. The Decision Making component then selects the most appropriate VO (and consequently RWO) for replacing the problematic VO. Information on the re-configuration of the CVO is stored in the CVO registry.

In order to obtain results on the efficiency and scalability of this framework a number of experiments was conducted for different number of available VOs in the area of interest. Figure 6 presents the time required for the Decision Making mechanism to find the optimal composition of VOs, as the number of available VOs increases. As can be observed, while this time increases as the number of VOs increases it remains lower than one second. The ILOG CPLEX OPTIMIZER (CPLEX algorithm) was utilized for the evaluation of this mechanism. Figure 7 depicts the execution times for all

operations. As it is expected, the *Dynamic Creation of a CVO* is mostly affected by the increase in the available VOs. In particular, over 700msec are needed for 10 VOs, while this amount is increased to over 7000msec for 50 VOs. Time savings are evident in the case of the *Knowledge-based Instantiation of a CVO*, where the requested service can be offered in just a few msec (due to the operation of the Request and Situation Matching mechanism only). Additionally, time savings are achieved in the case of the *Self-healing of a CVO* too. This operation replaces only the VO that cannot operate properly and exploits the remaining ones. Therefore, minimum time is needed, in comparison with the creation of the CVO from scratch.

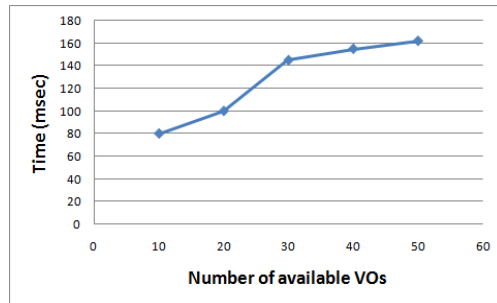


Fig. 6. Decision Making mechanism execution time

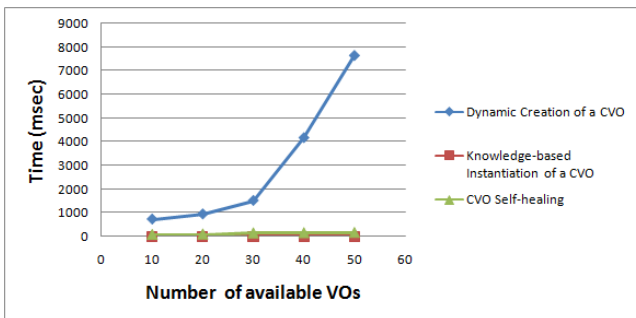


Fig. 7. Execution time of Cognitive management framework operations

4.2 Ambient Assisted Living Example

This sub-section presents an indicative example of how the three operations of the proposed framework can be applied in an Ambient Assisted living scenario. The scenario comprises two different (business) domains; (a) a smart home, where an elderly woman (Sarah) who has opted for an assisted living service lives in, and (b) a medical center, where doctors monitor Sarah’s environmental conditions and health status remotely, using the smart objects that exist in smart home. A set of RWOs have been installed in the smart home, comprising of sensors (temperature sensor, humidity sensors, luminosity sensor) that are connected to a Waspnote platform [15] and

actuators (namely a fan, a Light Emitting Diode (LED), and a lamp) that are connected to an Arduino platform [16].

First, a doctor through an appropriate user interface (Figure 8) provides a set of application/service requirements, functions and policies (e.g. energy efficiency). At the service level, the set of request and situation parameters are extracted and forwarded to the Request and Situation Matching mechanism, which, in turn, searches in the CVO registry for a previously created CVO that could fulfil the requested application/service. Initially, an appropriate CVO is not found, and the provided parameters are sent to the Decision Making, which will select the most appropriate VOs to satisfy the requirements and policies in the best possible way, and will trigger the creation of a new CVO. The newly created CVO is registered in the CVO registry together with the situation parameters under which it was requested for future reference by the Request and Situation Matching. The doctor, or member of the medical staff, can use the dynamically created CVO to monitor the medical status of Sarah. Feedback regarding the operation of the CVO can be provided and be stored in the CVO registry for future use. Sometime later, another doctor issues a similar request for monitoring Sarah. This request is elaborated at the service level and the derived request and situation parameters are forwarded to the Request and Situation Matching. This time, an already composed CVO that can fulfil the application/service requirements exists in the CVO registry; therefore the Request and Situation Matching proceeds with its immediate instantiation, without the need of triggering the Decision Making mechanism. During the execution of the CVO, one of the used luminosity sensors fails. This is detected by the implemented CVO and the Decision Making mechanism is triggered so as to find alternative luminosity sensor devices that could be exploited (through their corresponding VOs) in order to replace it in the overall CVO, thus ensuring continuity of the service delivery. The CVO is appropriately re-configured.

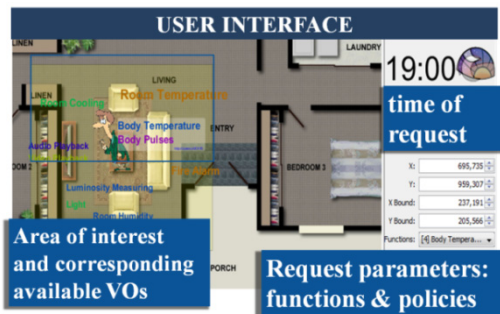


Fig. 8. Graphical User Interface

5 Conclusion

This work described a cognitive management approach to address the huge amount of data, devices and services which the Internet of Things will be comprised of.

The proposed Cognitive Management framework aims to conceal the technological heterogeneity, comprise the perspective of users/stakeholders, and facilitate situational awareness, higher reliability and energy efficiency. The framework comprises three levels of reusable cognitive components, namely the VO level, the CVO level and the User/Stakeholder and Service level. Thus the proposed Cognitive Management framework acts as an open networked architecture that can remove the sector specific boundaries of the early realisations of IoT, and enable a new range of Internet enabled services, which can be integrated with enterprise business processes. Through these features, and by doing work above the levels of the real-world and digital objects, the proposed framework constitutes a unique approach for the efficient integration of the IoT into the service layer of the Future Internet. The paper also presents a first indicative implementation of the proposed framework, comprising real sensors and actuators. The preliminary results of this work demonstrate high potential towards self-reconfigurable IoT.

Acknowledgment. This article describes work undertaken in the context of the iCore project, ‘Internet Connected Objects for Reconfigurable Ecosystems’ (<http://www.iot-icore.eu/>). iCore is an EU Integrated Project funded within the European 7th Framework Programme, contract number: 287708. The contents of this publication are the sole responsibility of iCore project and can in no way be taken to reflect the views of the European Union.

Open Access. This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

References

1. Uusitalo, M.: Global Vision for the Future Wireless World from the WWRF. *IEEE Vehicular Technology Magazine* 1(2), 4–8 (2006)
2. Weiser, M.: The Computer for the Twenty-First Century. *Scientific American*, pp. 94–10 (September 1991)
3. FP7-ICT-287708 project iCore (Internet Connected Objects for Reconfigurable Ecosystems), <http://www.iot-icore.eu> (accessed February 2012)
4. Giusto, D., Iera, A., Morabito, G., Atzori, L., Blefari Melazzi, N.: CONVERGENCE: Extending the Media Concept to Include Representations of Real World Objects. In: *The Internet of Things*, pp. 129–140. Springer, New York (2010)
5. Castellani, A.P., Bui, N., Casari, P., Rossi, M., Shelby, Z., Zorzi, M.: Architecture and protocols for the Internet of Things: A case study. In: *Proc. Pervasive Computing and Communications Workshops (PERCOM Workshops)*, pp. 678–683 (March 2010)
6. FP7-ICT-215923 project SENSEI (Integrating the Physical with the Digital World of the Network of the Future), <http://www.sensei-project.eu> (accessed February 2012)

7. Kostelnik, P., Sarnovsky, M., Furdik, K.: The Semantic Middleware for Networked Embedded Systems Applied in the Internet of Things and Services. In: Proc. 2nd Workshop on Software Services (WoSS), Timisoara, Romania (June 2011)
8. ARTEMIS SOFIA project, <http://www.sofia-project.eu/> (accessed February 2012)
9. Franchi, A., Di Stefano, L., Tullio, S.C.: Mobile Visual Search using Smart-M3. In: Proc. IEEE Symposium on Computers and Communications (ISCC) (June 2010)
10. Ubiquitous ID Center, <http://www.uidcenter.org/> (accessed February 2012)
11. 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: Architecting the Internet of Things. Springer (December 2010)
12. De Poorter, E., Moerman, I., Demeester, P.: Enabling direct connectivity between heterogeneous objects in the internet of things through a network service oriented architecture. EURASIP Journal on Wireless Communications and Networking, 61 (2011)
13. Kelaidonis, D., Somov, A., Foteinos, V., Poullos, G., Stavroulaki, V., Vlacheas, P., Demestichas, P., Baranov, A., Rahim Biswas, A., Giaffreda, R.: Virtualization and Cognitive Management of Real World Objects in the Internet of Things. In: Proceedings of The IEEE International Conference on Internet of Things, iThings 2012, Besancon, France (2012)
14. Schonwalder, J., Fouquet, M., Rodosek, G., Hochstatter, I.: Future Internet = content + services + management. IEEE Communications Magazine 47(7), 27–33 (2009)
15. Waspote, <http://www.libelium.com/products/waspote> (accessed February 2013)
16. Arduino, <http://www.arduino.cc/> (accessed February 2013)