An ontological model based on the ontology driven architecture paradigm for a middleware in the management of nano-devices in a smart environment

A Lopez-Pacheco\textsuperscript{1}, J Aguilar\textsuperscript{1}, E Puerto\textsuperscript{2}, and R García\textsuperscript{3}
\textsuperscript{1}Universidad de Los Andes, Mérida, Venezuela
\textsuperscript{2}Universidad Francisco de Paula Santander, San José de Cúcuta, Colombia
\textsuperscript{3}Universidad de Sinú, Montería, Colombia

E-mail: aguilar@ula.ve, eduardpuerto@ufps.edu.co

Abstract. In this paper is proposed an ontology framework for an autonomic reflective middleware for the management of nanodevices, which is based on the ontology-driven architecture paradigm. In this paper are presented the different levels of the ontology driven architecture model. At the first level, the computation independent model level defines the different ontologies of the nanotechnology domain, like a taxonomy of nanodevices, among others. The second level is the platform independent model level, where are defined the different ontologies for the deployment of the platform, which are the autonomic computation for the autonomous behavior of the middleware, data analytics that describe the analysis process of nodata, and service-oriented architecture because the autonomic reflective middleware is defined as a multi-agent system oriented by services.

1. Introduction
Nanotechnology is relevant in the ambient intelligence (AmI) because has features, such as real time monitoring, and notable sensitivity, flexibility and robustness. These characteristics introduce a real revolution at dissimilar areas of daily life, like health, military, transportation, among others [1-3]. Therefore, the nanostucture interconnection opens a new mechanism for the actuation in an AmI, in order to respond to different contexts [4]. For instance, nanosensors have fulfilled the gap in providing a full system characterization, which is not size dependent anymore, due to nanodevices capabilities and properties.

The utilization of the nanotechnology in an AmI requires of a middleware for the management of the nanodevices. In previous works has been proposed the architecture of an autonomic reflective middleware for the management of nanodevices for AmI, called ARMNANO [5]. ARMNANO is assembled in layers that contain the functionalities of the architecture in order to deliver its services. Some of the layers are a level of nanodata analysis [5,6], a level for the management of nanocommunications [7], among other levels.

ARMNANO requires the definition of all the knowledge of the nanotechnological context. The goal of this paper is the definition of the ontology framework of ARMNANO, based on the ontology driven architecture (ODA) paradigm [4]. The ODA paradigm aims to outline a common ontological architecture to place information in a generic platform. It proposes the use of ontologies to specify the logical ordering in an organization [8]: i) computation independent model (CIM): it is focused in the system domain, as well as in the functional requirements, non-functional properties, business rules,
structure goals, and data processing strategies that the system must follow; ii) platform independent model (PIM): through this model, the paradigm includes the details for the implementation at real conditions of the system; iii) platform specific model (PSM): this model combines the PIM with the proper details and features of the deployment platform.

In ARMNANO, an ontological framework is very important in order to describe the nanocontext, the nanoservices, etc., to integrate and to interoperate the different nano-components of the AmI [9-11]. The ontology framework of ARMNANO is divided in different models, such as the CIM, where is defined the domain ontologies; and the PIM, where are defined the different paradigms used to define the middleware (autonomic computing, service-oriented architecture (SOA) applications, etc.).

The main contribution of this article is that it describes the knowledge model required for future developments in nanotechnology-mediated environments, such as an AmI. The works in the literature normally propose very specific aspects for the use of nanotechnology, such as those related to communication problems [2-4,7] authentication of nanodata [1,6,12], among others, but do not study the knowledge required for the use of the nanotechnology. Also, ontologies have been defined, but without indicating how they can be integrated into a middleware such as ARMNANO [9-11].

2. ARMNANO middleware
ARMNANO is a multilayer architecture that provides services for nanodevices in an AmI [5]. The layers that belong to ARMNANO are distributed in two levels. The base level and the meta level of ARMNANO have a transversal level, called the data analysis smart system (DASS), which is in charge of performing the data analysis tasks using nanodata. The base level contains the physical devices in the AmI, such as the nanodevices and the microgateway. The abstract views of the nanodevices are deployed as agents, called NaS and NaA. Finally, ARMNANO comprise 8 layers (based on the works [13,14]), which are briefly described (for more details, see [5]): i) nanosensor and nanoactuator physical layer (NSAPL): it describes the nanosensors, nanoactuators, nanorouters and microgateway in the AmI; ii) nanosensor logical management layer (NSLL): It is composed of the logical abstractions of the nanosensors in the AmI, defined as NaS agents, with software for the authentication of the sensed data; iii) nanoactuator logical management layer (NacLL): It holds the NaA agent, which represent the nanoactuators in the AmI. Their logical functions include executing the commands and sending feedback up to DASS; iv) multiagent system management layer (MMAS): defined by a set of agents standardized by IEEE, for the management of the agents of ARMNANO; v) communication management nanoagent (CmNA): it addresses the communication and authentication protocols for the nanodata. It has been defined in detail in [7]; vi) service management layer (SML): it connects the multiagent system and SOA paradigms. It is a crucial layer to deploy web services. It has been defined in detail in [13]; vii) context awareness layer (CAL): it deploys context-based services, such as context discovery, modelling and reasoning; viii) ontological emergence layer (OEL): it deploys services for the emergence of ontologies (see [14] for more details).

3. ARMNANO Ontology

3.1. Computation independent model layer in ARMNANO ontology
The CIM layer defines the capabilities deployed by ARMNANO, its functional and non-functional requirements, and a taxonomy around the nanotechnology concepts that is split according to different aspects in this field (see Figure 1). Nanotechnology has features that fit several categories signaled in Figure 1: i) material: it signals the composition and morphology of the nanoobjects; ii) function: nanodevices can be nanosensors, nanoactuators and nanorouters; iii) strategy: it deals with the nanotechnology approach to generate surface modification, specialized materials, patterns of solids, among other strategies; iv) communication mode: here is specified the options for the data transmission at the nanoscale (see [7] for more details); v) context: ARMNANO can be used in different fields, for a variety of contexts, such as health, military or transportation; vi) morphology: the nanomaterials have a variety of forms with different aims and properties. In our ontological model, the main relationship
among the concepts is "is-a". In general, the nanotechnology knowledge at the CIM level is key for ARMNANO, due to it defines the context awareness.

Figure 1. CIM layer of the ARMNANO ontology.

3.2. Platform independent model layer in ARMNANO
The design of ARMNANO is based on three paradigms, which are specified as ontologies (Figure 2): data analytics, SOA services and autonomic computing. The relationship among these ontologies are: SOA “calls” data analysis to give services for the nanodata processing, data analysis “feeds” the autonomic computing because the cycles use it, and finally, an autonomic computing (AC) device “delivers” its tasks using the SOA capabilities.

Figure 2. PIM layer of ARMNANO ontology.

3.2.1. Data analytic ontology. In this ontology is conceptualized the different aspects included in a data analysis process (see Figure 3) [15,16].

- Concepts: the concepts/classes of the data analytics ontology are; i) data analytics: it relates to the deployment of specific mathematical or statistical approaches to different types of nanodata processing. ii) techniques: it refers to specific theoretical methods deployed to analyze the nanodata of interest. iii) tasks: it defines the gender of data analysis to be designed. iv) objectives: it points out the goal of a specific task of data analysis. v) types: in the context of ARMNANO, a task of data analysis can be used to monitor, to analyze or to plan actions in an Aml.
• Relations: it outlines the kind of connection between the concepts, which are; i) data analysis has a technique, ii) data analysis is a task, iii) data analysis has an objective, iv) data analysis is a type (see Figure 3).
• Instances: it defines the possible instantiations of the concepts; i) techniques: neural networks, linear regression, etc.; ii) tasks: they include the classification, prediction, clustering, among others; iii) types: they refer to the different kind of services delivered in the data analytics ontology for an AC device.

![Figure 3. Data Analysis Ontology for ARMNANO architecture.](image)

3.2.2. Service-oriented architecture services. This ontology includes the conceptual items, relations among these concepts, and instances involved in a given context, to deploy SOA applications (see Figure 4) [17,18].

• Concepts: it defines the main concepts in a SOA context: i) languages: it points out the different languages to define service applications; ii) service: it defines the services that can be offered in ARMNANO through its architecture. iii) repository: is referred to where the services are stored; iv) web service: it points the fact that the service is optionally provided on the web; v) standard: it defines the standards used for the development of a SOA application; vi) provider: it refers the external provider of services; vii) consumer: it is the direct beneficiary of the service; viii) bus of services: is the platform used for the management of the services.
• Relations: i) repository have services: the repository saves the services that ARMNANO provides; ii) web service is a service; iii) web services has standards; iv) services have languages; v) a service has a provider, vi) a consumer has providers; vii) a bus of services handles services.
• Instances: some examples of instances are: i) standards: for example, service oriented architecture protocol (SOAP), and universal description, discovery and integration (UDDI). They are called depending on the architecture requirements; ii) languages: it depends on kind of data and context where this data is treated. For example, web semantic language (WSDL), extensible markup language (XML), and representational state transfer (REST).

3.2.3. Autonomic computing. Is one of the main paradigms of the ARMNANO design [5,15,19]. The autonomic computing ontology defined for ARMNANO, is characterized at Figure 5.

• Concepts: it includes the concepts about the autonomous behavior of ARMNANO: i) managed entities: it relies on the objects considered in the AmI; ii) controller: it defines the different tasks for the management of the objects in the AmI. It is based on the MAPE+K loop concept [15-20]; iii) act: is the action to be executed in the AmI; iv) application: software used by the tasks
of the autonomic cycle; v) object: it describes the hardware and software managed in the context of the AmI; vi) person: is an external actuator, either a consumer or provider to ARMNANO; vii) device: it is one specific hardware in the AmI; viii) monitor: represents the tasks to track the variables that define the context; ix) analyze: represents the tasks to perform the nanodata interpretation, x) execute: represents the commands executed in the AmI; xi) plan: represents the tasks to schedule.

- Relations: i) managed entities are an object; ii) managed entities are a person; iii) object is a device; iv) object is an application; v) controller has to monitor; v) controller has to analyze, vi) controller has to plan; vii) controller has to act; viii) controller handles managed entities; ix) monitor is an application; x) plan is an application; xi) act is an application. (see Figure 5).

Figure 4. SOA Ontology for ARMNANO architecture.

Figure 5. Autonomic Computing Ontology for ARMNANO architecture.

4. Case study

4.1. Context

To test the ontologies, herein is described a case study: “Poly-traumatism patient due to a car accident” [12]. In this case study, a car driver suffers an accident, thus, he enters the first care aids at the closest hospital. The driver has severe damage in the arms and head. In the smart hospital is deployed ARMNANO, and the context is characterized by external nanosensors and in situ injected nanosensors into the patient. There are 15 external nanosensors, meanwhile the patient is injected locally at the affected zones with nanosensors, nanoactuators, nanorouters, with the following distribution: 100 nanosensors (NS), 100 nanoactuators (NA) and 30 nanorouters (Ro). There are 15 NS, 15 NA and 3 Ro actives. Rest of NS, NA and Ro are activated just if there is a failure in one or several of the starting nanodevices. In this context are monitored two variables, histamine level (to know the inflammation level), and the hemoglobin concentration (to know the amount of lost blood).

4.2. Computation independent model ontological definition

In this case study in the health area, the instantiation of the CIM ontology is shown in the Figure 6. It defines the domain knowledge in this context. This ontology signals that the “material” of the nanodevices are carbon-based materials and semiconductors. The carbon-based structures are graphene and carbon nanotubes (CNT). The semiconducting properties are instantiated for the construction of data processors. They are the nanosensors or the nanoactuators. In the function branch are defined three functions: sense, actuate and route. The “strategy” for these nanostructures relies in surface patterns and/or chemical and physical deposition, meanwhile the communication mode at the nanoscale is electromagnetic or molecular.
4.3. Platform independent model ontological definition

In the PIM ontology, for this case study, for the “data analysis” ontology, the “techniques” applied relies on the neural networks, used to process the nanodata in order to detect changes in the targeted variables (see Figure 7). Additionally, the autonomic computing ontology outlines that the tasks are classification, optimization and prescription. Now, it is given an example of the definition of these tasks with our ontology: i) classification: is an application, which is used by the monitoring phase to control the patient. This application splits the nanodata in classes, based on the targeted variables; ii) optimization: is an application, which is used by the monitoring phase to improve the conditions of the patient; iii) Prescription: is an application, which is used by the planning phase to define the actions exerted in the patient. They are part of the controller. The PIM level defines the rest of the things required by ARMNANO to monitor, analyze and plan the context. The rest of the concepts for the data treatment in this case study, are: the platform is deployed for data transmission with the protocols HTTP, XML, WSDL, and UDDI. In this way, the services offered by ARMNANO can be delivered directly, or via a web interface. Thus, an external user, such as medical personnel, can use the saved data for offline analysis, and ARMNANO is devoted to measure and report, when requested, the variables of interest: histamine and hemoglobin.

![Figure 6. CIM layer for the case study.](image6.png)

![Figure 7. Data analytics ontology for the case study.](image7.png)

5. Conclusions

Herein have been defined the capabilities of ARMNANO respect to the ontological description of the context. For this purpose, the ontological model of ARMNANO has been supported in the CIM and PIM levels of ODA paradigm. The CIM layer defines the domain concepts around the ARMNANO middleware. Its domain ontology is anchored in the nanotechnology area, but with a general description in order to allow that ARMNANO can be deployed in generic contexts of AmI based on nanodevices. At the PIM layer are defined the three areas that describe the instantiation of ARMNANO in real applications. The areas are the autonomic computing, SOA capabilities and data analysis. All of them are interconnected due to be complementary. The case study shows this specification in the context of a car driver accident. The injured person was treated using the ARMNANO capabilities. The CIM and PIM model aid to describe this case.

This research has allowed showing the ARMNANO potential to provide answers with autonomy, which talks about its robustness, versatility and effectiveness. Its ontological framework allows the contextualization of ARMNANO in different contexts. Next publications must be about the implementation of the different elements of ARMNANO, particularly in the CmNA and DASS levels.
References

[1] Akyildiz I and Jornet J 2010 The internet of nano things IEEE Wireless Communications 7 58
[2] Dressler F and Fischer S 2015 Connecting in-body nanocommunication with body area networks: challenges and opportunities of the internet of nanothings Nano Communication Networks 6 29
[3] D’Oro S, Galluccio M, Morabito G and Palazzo S 2015 A timing channel-based MAC protocol for energy efficient nanonetworks Nano Communication Networks 6 39
[4] Alkyildiz I, Jornet J and Pierobon M 2011 Nanonetworks: a new frontier in communications Communications of the ACM 54 84
[5] Lopez P and Aguilar J 2015 Autonomic reflective middleware for the management of nano devices in a smart environment (ARMNANO) Current Analysis on Communication Engineering 2 56
[6] Lopez P, Aguilar J, Pinto A and Torres J 2018 Data analysis smart systems in a nanodevices-based middleware Contemporary Engineering Sciences 11 4665
[7] Lopez A and Aguilar J 2018 NANO-Communication management system for smart environments REVECOM 5 12
[8] Aguilar J and Fuentes J 2014 Management system of learning paradigms using ODA Proc. XL Latin American Computer Conference (Uruguay) 1
[9] Sharma A 2016 Nanotechnology ontology: semantic access to information in the nano world Journal of Library and Information Technology 36 29
[10] Taqnaka M 2005 Toward a proposed ontology for nanoscience Proc. of the Annual Conference of CAIS 1
[11] Mizoguchi R, Kitamura Y and Kozaki K 2013 Systematization of nanotechnology knowledge through ontology engineering Citeseer 1
[12] Rodrigues J, Paulovich. F.de Oliveira M and de Oliveira O 2016 On the convergence of nanotechnology and big data analysis for computer-aided diagnosis. Nanomedicine 11 959
[13] Aguilar J, Jerez M, Mendonca M and Sanchez M 2016 MiSCI: autonomic reflective middleware for smart Cities Communication Computing and Information Science Series 658 241
[14] Aguilar J, Jerez M, Sánchez M and Mendonca M 2017 Emergencia ontológica basada en análisis de contexto, como servicio para ambientes inteligentes Revista DYNÁ 84 28
[15] Aguilar J, Cordero J and Buendia O 2018 Specification of the autonomic cycles of learning analytic tasks for a smart classroom Journal of Educational Computing Research 56 866
[16] Aguilar J, Sánchez M, Cordero J, Valdiviezo-Díaz P, Barba-Guamán L and Chamba-Eras L 2018 Learning analytics tasks as services in smart classroom Universal Access in the Information Society 17 693
[17] Granell C, Llidó D, Berlanga R and Gould M 2014 An ontology driven architecture for semantic web service composition Journal of Network and Computer Applications 6
[18] Mezni H and Sellami M 2015 AWS-Ont: an ontology for the self-management of service-based systems Proc. IEEE 8th International Conference on Service-Oriented Computing and Applications 85
[19] Vizcarrondo J, Aguilar J, Exposito E and Subias A 2012 ARMISCOM: Autonomic reflective middleware for management service composition Proc. 4th IEEE Global Information Infrastructure and Networking Symposium, IEEE Communication Society 1
[20] Vizcarrondo J, Aguilar J, Exposito E and Subias A 2017 MAPE-K as a service-oriented architecture IEEE Latinoamérica Transactions 15 1163