TOWARDS A ONTOLOGY DEFINED ENVIRONMENT

William C. de Souza Carvalho, Pedro Frosi Rosa, Flávio de Oliveira Silva
Universidade Federal de Uberlândia
william.carvalho@ufu.br, frosi@ufu.br, flavio@ufu.br

Abstract: The aim of this paper is delineating some core fundamentals of a proposal for achieving semantic interoperability in the distributed system field by defining a topology of ontological repositories linked to an ontology defined environment that mediate the exchange of services between network elements that uses ontology as language to describe both user requirements and providers capabilities. The paper describes the state of art of the ontology uses in distributed systems, some actual internet architecture limitations and a brief approach oh the environment definition method including some of its layers. The environment was proposed in a scenario where the ETArch future internet architecture components were used as conceptual background to illustrate supports the needs of multimedia applications.

Keywords: Ontology, Development Environment, Future Internet

1. INTRODUCTION

In the Biblical story of the Tower of Babel, God caused the failure of the tower-building project when interrupted communication among workers by creating various languages. In the modern history of Information Technology, several languages have been created to meet a wide range of organizational needs through a myriad of methods, languages, paradigms, protocols, frameworks, environments and processes. As in the biblical history, the diversity of languages, protocols, topologies, etc. has caused confusion and misunderstanding; although each in its own way intends to reverse this situation by ensuring the advent of a redemptive vision of knowledge integration.

Since the 1970 years the Internet has become much more than architecture in the strict sense once it enormous success was responsible for a global telematic revolution and a complete turnaround in mankind history. It has become a concept that has revolutionized the way people communicate and share knowledge and things (BROWN, 2006), (WALDROP, 2008); which in the Internet Age (IoT - Internet of Things) may be different in nature, essence, or breadth (SANTUCCI, 2010). In this period, the great technological evolution around the Internet was accompanied by something so important or more important - the cultural revolution regarding the sharing of information (BISBY, 2000).

Despite all the diversity of applications developed over these decades, the essence of the Internet architecture remains unchanged (ALMAGOR, 2011), (SHERRY, 2011), (TRONCO, 2010) e (LEINER, 2009). TCP/IP protocols received small changes around the beginning of the 1980s, but essentially remain the same initially proposed (FOROUZAN, 2008).

In its early days, the Internet witnessed the advent of two of the most widely accepted applications: file transfer, for data sharing (SHERRY, 2011) e (rfc959); and, electronic mail, for interpersonal interactions (SHERRY, 2011) e (rfc821). At that time, there was no talk of time constraints nor demands for high resource availability such as current voice and video conferencing applications have (NYGREN, 2010) e (CIULLO, 2008).

Considering that current networks have flow rates of more than millions of bits per second (Mbps), some questions may arise: "Why do voice applications (VoIP - Voice over IP) require only 56 kbps do not function properly in significant part of the time?"

In the 1960s and 1970s, predictable requirements for networks were limited by the role of existing applications and also by the technologies and capabilities available (Almagor2011). At that time, available throughput did not exceed 2,400 bps (Leiner2009). The central protocols of the Internet Architecture, TCP / IP, were developed against the backdrop of the described reality.

Nowadays, long-distance links implemented via optical fibers deliver flows greater than 2 Tbps. However, this capability can’t be fully and adequately exploited by current applications because the Internet Architecture transport and network layers function as a gap} that separates the upper layers (applications) from the lower layers (links). The interface of the TCP and UDP transport protocols does not provide any mechanism to translate an application's Quality of Service (QoS) requirements and map them to link capabilities.

Networks are designed following architectures, as is the case of the Internet, whose layers and respective protocols are defined a priori and do not fit the needs of the applications, that is, once the network configuration, regardless of the application, it will always work the same way.

The Internet Architecture does not distinguish applications, QoS or QoE, treating everyone the same way (BE - Best Effort}), this being the main reason why many applications are used far below their potentialities. As exemplified above, in many communications, the network layer (IP Protocol) causes unnecessary overhead. In this case, the environment defines which layers, protocols, requirements and capacities are appropriate for the moment.

Another major challenge to organizations is the difficulty of interconnecting their systems. That occurs because the upgrade - even minimal - of the topology of the solution, architecture and or technologies can make the cost of these changes costly or time-consuming (Choi2006, Chen2009). In addition, the heterogeneity of information system elements makes it difficult for companies to respond to market requirements, negatively impacting their competitiveness.

Even when communication happens on a local network, distinct users may require different network requirements, e.g., in the case of two users who want to watch the same show, but one of them makes use of FULL HD digital TV, with 4k transmission, while the other wants to see the show from a smartphone. In this case, the required bps is totally different for them.

Ontologies are widely used in knowledge engineering, artificial intelligence and computer science (BALDAUF, 2007). They are present in applications of the most varied domains, such as knowledge management, natural language processing (BATEMAN, 2010), e-commerce (WEILONG, 2008), information retrieval, (PARALIC, 2003), database integration (ALALWAN, 2009), bioinformatics (BAKER, 1999), education (VAS, 2006), health (KIONG, 2011), agriculture (SU, 2012) and IoT (HACHEM, 2011).

There are several requirements that applications may require from flow, timing, latency, security, reliability, accuracy, mobility, etc., which may require configurations that change over time. In addition to this diversity of requirements, a strictly user-related aspect of the user experience (QoE - Quality of Experience), that is, a certain QoS requirement can vary in amplitude according to the user.

The absence of a reference environment for the development of interoperable applications able to handle this range of possibilities, coupled with the myriad of network protocols are critical factors that put organizations facing increasing challenges to integrate their sources of information (CHEN, 2009a), (FARQUHAR, 1997) e (FIKES, 1999). IEEE considers an interoperable system or product to be able to interact with other systems or products without any special effort by a customer (XU, 2011).

So, the creation of an Ontology Defined Environment would allow the network to be able to easily meet user experience requirements and its interactions with the distributed environments considering the diversities mentioned above.

The aim purpose of this paper is to present the main lines of a proposal to define a topology of ontological repositories linked to an ontology - defined environment that meets the requirements of distributed environments and supports the needs of multimedia applications. Some justifications and contributions deriving from the use of ontological repositories in distributed environments are:

- Investment Reduction: The specification of the protocol stack no longer resides on the network elements and is stored in a repository; which means that the introduction of a new protocol no longer means investments in the network;
- Maintenance Cost Reduction: evolution activities, bug fixes, protocol updates or new services launch will only focus on the ontology repository, which reduces maintenance costs;
- Agility: currently the launch of new services depends on the change of several network elements. Using the centralized repository will reduce maintenance cost and time; which assures faster product availability and Resource Optimization:

The use of ontology will help improve system interoperability through its inherent semantic heterogeneity (SUN, 2011). Even if information systems use the same syntax, they can associate different meanings to things, which prevents the exchange of information. With ontologies, it will be possible to unambiguously specify the underlying vocabularies of information systems (FARINELLI, 2013) e (BAJWA, 2011).

2. CONCEPTS REVIEW

Although ontology has a clear definition in philosophy, there is substantial terminological confusion when used in the different areas of computer science about what the term denotes. Thus, it is necessary to clarify the terminology and establish the formal characterization of how the term is used in here; besides dealing with the relation between the definition of ontology and the notions of conceptualization and language.

Etymologically, *ont* comes from the present participle of the Greek verb *einai* (to be) and the word *logos* (knowledge). Hence, ontology (ont- + logos) can be understood as the “study of being”. The term, however, was popularized in philosophical circles only in the eighteenth century with the publication in 1730 of the *Philosophia prima sive Ontologia* by Christian Wolff (GUIZZARDI, 2005).

Despite historical divergences about the term’s paternity, there are records showing that it was coined in the seventeenth century by both the philosophers Rudolf Göckel in his work *Philosophicum Lexicon* and Jacob Lorhard in his *Ogdoas Scholastica* (GUIZZARDI, 2005).

In the earlies of the seventeenth century, Clauberg and Jacobus Thomasius used the world Ontology to denote the terms *First Philosophy* or *Metaphysics*. Clauberg’s definitions for Ontology encompass traditional knowledge of metaphysics, but in a more formal way. Thus, several theorists have adopted Ontology as a definition for “general metaphysics”.

In other hand, a distinct sense for Ontology was given by Stanisław Leśniewski who called it his logic when he presented his system of calculating names, with the separation of Protothetic Propositional Calculus; Mereology Algebra of Classes, Except the Null Class; and, Ontology Theory of Classes and Relations. According to c, ontology should be the basis for the formalization of logic, with little relation to classical ontology. However, Tadeusz Kotarbiński and Leon Chwistek have indicated that Leśniewski’s calculations have a very close relation to

Aristotle's formal logic and therefore have the same traditional bases. Leśniewski confirmed this position with his axiomatic ontology, taking it as the basis for the formalization process.

The main aim of ontology is to study the more general characteristics of reality and real or imaginary objects (PEIRCE, 1960), that is, the study of the generic characteristics of each mode of being (SMITH, 2008) and (SMITH, 2003}. Unlike various specific scientific disciplines such as physics, chemistry, and biology, which deal only with entities that fall within their respective domain, ontology deals with relations between categories, including relationships that occur between entities belonging to domains distinct from science, and also by entities recognized by common sense (BARCELLOS, 2009), (HARTMANN, 2009) and (GUIZZARDI, 2005}.

Ontologies enable the development of theories that deal, for example, with persistence and change, identity, classification and instantiation, causality, etc. According to (CORAZZON, 2003), (CARVALHO, 2013) and (HARTMANN, 2009), ontological questions include questions such as: what kinds of entities are there? What is the difference between events and objects? How such things are related? What are the properties of a thing and how do they relate to the thing itself? What is the essence of an object? Does essence precede existence? Is an object equal to the sum of its parts? These are general, but factual, and fundamental issues for science, regardless of whether the subject is the properties of atoms, human organs or insurance requisites, or even if the goal is to develop theories of physical, mental, or social events (CORCHO, 2006) and (Hartmann, 2009).

Other philosophers such Herbert also used the name Ontology for the sciences of reflection and not as an intuitive science. Husserl separated this term into *formal* or *material*, considering the issues of material ontology and ontology formal (SMITH, 2007) and (CORCHO, 2006).

In the axiomatization of scientific theories some ontological concepts appear explicitly: part, composition, system, relations, limit, causality, state, event, change, property, right, possibility, process, space and time. However, the specific axioms of these theories generally say nothing - or very little - about these fundamental and generic concepts (GOMEZ-PREREZ, 2004).

The sciences simply borrow and use them in an intuitive, informal and pre-systematic state. Although these generic concepts are common to various sciences, no single scientific discipline goes to the trouble of putting them together in a single body (BUNGE, 1977). The same happens with the use of these concepts in computer science and particularly in conceptual modeling. Concepts such as *part* and *all*, *instantiation* and *sorting*, *assignment* and *relations* and *causality* are represented by primitives of several conceptual modeling languages or, at least, are used in the discourse of the Computer Science literature.

In the early twentieth century, Edmund Husserl coined the term Formal Ontology as an analogy to Formal Logic. While Formal Logic deals with formal logical structures such as truth, validity, and consistency, regardless of their truthfulness, Formal Ontology deals with ontological structures such as parts theory, set theory, types and instantiation, identity, dependence, and unity, with the formal aspects of objects regardless of their particular nature.

The development of formal ontologies as a philosophical discipline aims at the development of a system of general categories. This system can be used in the development of scientific theories and theories of reality based on knowledge of specific domains. While scientists deal with specific issues, ontologies deals with cross-cutting issues in all domains.

Nowadays, several ontological systems were constructed in projects related to computer science (MASOLO et al., 2003) and (HELLER and HERRE, 2004). In this paper we consider only the formal ontological theories that can be developed and applied in the solution of problems in the areas of informatics and information sciences, and in particular, conceptual modeling.

> An ontology is an explicit and formal specification of a shared conceptualization. Conceptualization refers to an abstract model of some phenomenon in the world by identifying the relevant concepts of such phenomenon. Explicit means that the type of concepts used as well as the restrictions on their use are explicitly defined. Formal refers to the fact that the ontology must be interpretable by the machine. Shared reflects the notion that an ontology captures consensual knowledge, that is, it is not specific to an individual, but accepted by a group (Studer1998).

In the computer science field, ontologies must be encoded in a machine-interpretable language (GOMEZ-PREREZ, 2004) and (STUDER, 1998). Moreover, the perspective of an engineer about ontologies usually are more specific than for a philosopher. Finally, due to the use of the term ontology in computer science, ontology definition in this context consider characteristics of reuse and sharing, which are not essential in philosophical ontologies. Otherwise, Neches (1991) gave another definition, focusing on the form of an ontology:

> An ontology defines the basic terms and relationships that make up the vocabulary of a thematic area, as well as the rules for combining terms and relationships in order to define vocabulary extensions.

There are several formalisms and knowledge representation languages that can be used to formalize and implement ontologies. Each of them consider different kind of components which can be used for those functions. However, they share the following minimum set of components.

2.1. Class

Classes represent concepts in the broad sense. For example, in the travel domain, concepts can be: places (cities, villages, beaches), lodging (hotels, hostels, lodges, camping) and means of transport (airplanes, trains, cars, boats, motorcycles and ships). In an ontology, classes are usually organized into taxonomies that allow the use of inheritance mechanisms.

It is possible to represent a taxonomy of entertainment venues (theater, cinema, concerts) or travel packages (economic travel, business trip). In a knowledge representation framework-based it is also possible to define meta-classes, which are classes whose instances are classes. They allows the creation of different gradations of meaning, since it is possible to establish different classes layers in the ontology.

2.2. Relations

Relations represent a kind of association between domain concepts. They are formally defined as any subset of the Cartesian product of *n* sets in the form of

$$R \subset C_1 \times C_2 \times \ldots \times C_n$$

Ontologies generally contain binary relations. The first argument is known as the domain of the relation, and the second argument is the contradiction. For example, in the binary relation **`localArrival`**, the concept of Travel is its domain and the concept of Local is its contradiction. Relationships can be instantiated from the knowledge of the domain. In order to express that **`flight JJ3242Feb-15-2016 arrives in Uberlândia`**, it is written **`localArrival JJ3242-Feb-15-2016 Uberlândia`**.

Binary relations are used to express the attributes of the concept. Generally, attributes are distinguished from relations because their contradiction is a type of data, such as *string*, *number*, etc., whereas the contradiction of relations is a concept. The following code sets the `numeroVoo` attribute, which is a *string*. It is also possible to express relations of greater predicate, such as “a road connecting two different cities”.

Formal axioms serve to model sentences that are always true (Gruber, 1993). Usually they are used to represent knowledge that can't be formally defined by other components. In addition, formal axioms are used to verify the consistency of the ontology itself or the consistency of the knowledge stored in the knowledge base. Formal axioms are very useful for inferring new knowledge. An axiom in the Travel domain would be that it is not possible to travel from North America to Europe by train.

2.3. Instance

Instances represent elements or individuals in an ontology. An instance of the concept `JJ3242` is **`flight JJ3242 that arrives in Uberlândia on February 15, 2016 and costs R$ 937,00`**.

3. STATE OF ART

In the last four decades computer networks have been projected from equipment (NE - Network Element) that encompasses the functionalities of the data plane and also the plan of control. This also includes those based on the Internet architecture. This type of NE defines how the behavior of the network will be in design time. Small configuration adjustments can be done at deployment time, or even in post-deployment time frame.

The Internet architecture's (TCP, UDP, etc.) and network (IP, ICMP, etc.) transport protocols introduce the existence of a gap between the application and link layers that does not allow an application to specify the requirements to function properly. This is one of the main reasons why multimedia applications doesn't work properly regardless the network payload. The other reason is a matter of Internet architecture which is designed to work in the best effort and therefore does not provide support for traffic that requires real time.

The circumstances in which an application is invoked can influence the original Quality of Service (QoS) parameters. They may vary for a variety of reasons, and the QoE (Quality of Experience) is one of the main ones. However, aspects such security, power consumption, mobility, type of access device, location, among others, can significantly change the requirements over time.

3.1. Ontology in Distributed Systems

This section presents an analysis of the observed Internet Architecture ontological deficiencies that make it hard to meet new applications requirements of the future Internet applications. Ontology has been extensively used in computing disciplines for more than 30 years (MEALY, 1967). It has recently been used in computer networks and distributed systems (BLAIR, 2011).

We can examine the ontology usage in distributed environments by two main perspectives: first, it's interesting the use Ontology to formally define distributed systems concepts; and, secondly, the integration of distributed repositories ontologies to compose an interoperable Ontological System. Ontology allows defining and specifying concepts in an abstract way which could be input into ontological repositories maintenance tools.

In distributed environments[1] it is necessary to separate the ontologies in domains, otherwise it will be impossible to maintain the ontology repository. The Internet architecture has become a *de facto* standard for distributed systems. Its structure is based on the Open Systems Interconnection (OSI) Reference Model, without using the Session and Presentation layers. Its main protocols are TCP, UDP and IP were defined three decades ago. This same architecture

[1] Distributed environments are composed by communication infrastructure, sensors, distributed applications, users etc.

has supported the expansion of computer networks such as the Internet itself and Next Generation Networks (NGNs), IMS (IP Multimedia Subsystems), sensor networks, computing in the cloud, IoT (Internet of Things).

This expansion was largely based on meeting people's communication needs, which reflects in applications requirements that aim to support them. However, despite the computational evolution of machines and devices, there has been no significant improvement in layers 3 and 4 of the Internet Architecture. News applications requirements are met by new specifications or adjustments in old fashioned existing protocols.

3.2. Distributed Systems Modeling

The systems development market is increasingly interested in distributed systems models that abstract underlying details about operating systems, network protocols, and languages. The reason for this interest lies in the standardization and simplification of integrations of these systems, even though in some cases abstraction implies more processing capacity.

The use of standard models allows the reuse of technologies and also reduces decoupling between systems. In this technological field a number of middleware technologies have been developed, such as CORBA (Common Object Requester Broker Architecture), JEE (EJB and JMS) and Web Services.

The task of distributed systems modeling according to Almeida is based upon a design methodology in which he introduces the notion of abstract platform and platform independence. His methodology also discusses project quality, design process, modeling languages for abstract platforms and suggests a framework that ensures platform independence (ALMEIDA, 2006).

In distributed systems modeling, the concept of an abstract platform presents similarities with Service Oriented Architecture (SOA), which aims to standardize integration and promotes component modeling and reuse. Philosophically, these components represent the parts of distributed systems, being a common subject for both the ontological theories of classes and relations and distributed systems modeling approaches.

An approach proposed in 2001 is the development of applications whose architecture is model driven, or Model Driven Architecture (MDA). It was proposed by the Object Management Group (OMG) and recognizes the importance of the models and their interactions in the process, making them the cornerstone of software development.

MDA states that the software development process must be driven by the system modeling activity, done in conceptual level, independently of any platform / implementation choice. Through transformations carried out in this conceptual model, new models are generated, with levels of abstraction every time more specific and linked to implementation. Thus, the final version of the system could be generated automatically, from specifications originally defined in the conceptual model. The constructed models must be based on some form of formal representation of knowledge to avoid ambiguities and allow their execution by machines.

3.3. An Internet Architecture Ontology

Ontology, as a form of explicit representation of a conceptualization, has not been and has not been used in the lower and intermediate layers of the Internet architecture in the same way as the outside in the application layer. As mentioned previously, in areas such as database, software engineering or artificial intelligence, the internet is used as a means of realization and its distributed communications, and for this, use some specific and well know protocols.

When there is a new application requirement that the Internet architecture does not support yet, adjustments are made in the existing protocols or, in many cases, new protocols are proposed to comply with the new user expectations. This is done to minimize the impacts on the installed base, however, making these adjustments that meet the requirements quickly prevails on the importance of maximizing the efficiency of the technology.

For example, Quality of Service (QoS) is not a requirement implicitly recognized by current protocols, nor is it self-configuring. The original design of the Internet architecture did not predict that this would be necessary whereas, in current applications, there is a gap between the quality of service requirements demanded by the applications and what network can actually provide. This is the reason why applications may not work properly despite the increasing computing power of the devices. This limitation reinforces the need to revise the current architecture.

From this point of view, some concerns arise about the real need of the Network and Transport layers existence, according to the TCP/IP protocols. Considering the adjustments and propositions implemented in the Internet architecture in recent decades to meet the new applications requirements, it would be best to rethink the roles, functionalities and responsibilities of the Data Link, Network, Transport, and Session layers for the new generations of Internet architectures. A coherent architectural view can meet these requirements without redundancy or overlapping responsibilities.

The use of an ontology lifecycle management environment based on an ontological repository that allows the creation, instantiation and reuse of services that meet the requirements of the applications executed in a common logical bus is essential for the creation of solutions that are appropriate to the evolution proposals of internet architecture. To this aim, it is important to analyze the capabilities and needs of the real world to create the specification of the conceptual model of this environment, to ontologically satisfy the requirements of distributed systems.

4. THE PROPOSAL

This section describes the method and the creation schemes of Domain Ontologies, defines the concepts in each domain and presents the architecture of the Ontology Orchestration layer. The creation of an ontology development environment takes as premise that the behavior of the distributed system is directed by aspects of the environment and that all of them are modeled by an ontology.

ETArch (SILVA, 2013) is a materialization of the Title Model (SOUZA, 2012) and served as proof of concept for the introduction of some conceptual elements for future network communications: separation of identification and addressing; separation of data and control plans; traffic aggregation multicast and workspace; and, vertical texting for wireless technologies.

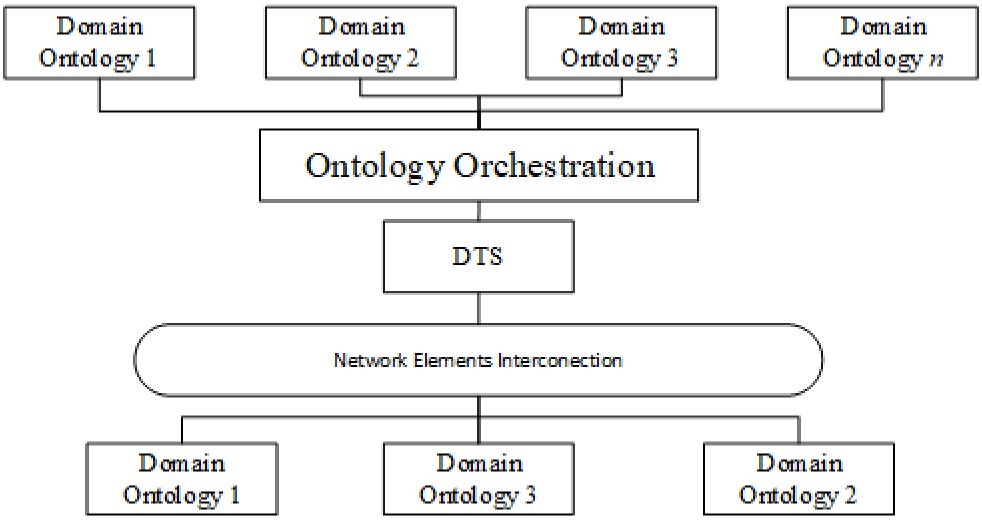

Figure 1 – Ontology Development Environment Architecture Overview

The next section will describe the main layers to be employed in the ontology development environment architecture.

5. ONTOLOGY DEVELOPMENT ENVIRONMENT LAYERS

Ontologies can be modeled in different layers or architectures, and generally, until today, they have been used restricted to applications (pereira_title_2011). In the field of distributed systems, particularly the computer networks area, the use of ontologies is not very common. In this section the ontology development environment required layers creation are specified from an architectural and philosophical perspective.

ETArch implementation solves aspects of network control such as DTSAs (DTS), Securities resolution, network elements configuration for workspace conformation, power consumption and entities mobility aspects. However, data plane has been relegated to its elementary use, i.e., the entities communicate through workspace using unconfirmed services.

5.1. Data Plan Architecture

The Data Link, Network and Transport internet architecture layers will be matched, with other layers defined in ontology terms. The definition of the communication context will be done through the Control Plan at times when an entity creates, changes or attach to some workspace.

5.2. Ontology Definition Layer

According to Noy (NOY, 2001) there are no correct way to model a domain because there are several possible alternatives. As a reliable alternative, a method is suggested with the following steps: Domain determination and Ontology Scope; consideration of the Reuse of Existing Ontologies; enumeration of Important Terms of the Ontology; class definition and class hierarchy; definition of Class Properties and Instance creation.

5.3. Ontology Integration Layer

The Domain Title Service obtains the modeled environmental definition rules from the Ontology Definition Layer, through the Ontology Integration Layer presented in Figure 1. This objective is to map the concepts defined in each of the domain ontologies and make them available to the Domain Title Service.

Each Domain Ontology will be represented by OWL in modules called Local OWL Ontology, specific to each area of knowledge, whose representations will serve as input to the ontology mapping function.

5.4. Other Layers

During the creation of the ontology development environment the work may be that other layers and ontologies are identified and described. At this point, we already have a sense of the needs of the creation of Domain Ontology and Security Vulnerability Ontology and the definition of Ontology Orchestration Layers, etc.

6. EXPECTED RESULTS

After creating this ontology development environment, it is expected that application requirements may use communication instances based on specific ontology protocols to provide the needed service. The term Internet means more than an architecture, it's a concept that allowed people to share their “things” naturally. Social networks are there to prove this concept.

ForCES (Forwarding and Control Element Separation) (rfc3746) shows that separation of control and data planes brings innumerable advantages to network environments because flexibility in reconfiguration of network elements allows significant provision time to new services implementation reduction (time-to-market).

A title domain service allows a workspace (i.e. the network environment) to be transparently reconfigurable to support the QoS / QoE requirements (including the mobility requirement) of the applications. In this way, it becomes a network operating system, whose interface can be used by agents that capture internal and external information, which will be translated into aspects of the network through its services. Moreover, to date, aspects such as security and charging, for example, have not been mentioned.

Various data captured by the applications must be translated into the network environment, otherwise it will not meet your requirements. For example, today's networks are capable of incredible flows, but a simple voice application does not work properly, depending on the moment. This occurs because today’s networks, particularly the Internet, are not able to handle the requirements.

In addition to the information coming from the environment, there are aspects of corporate policies that are overlooked by the settings of the network environments. For example, security policy is practically implemented in firewalls, which acts on specific points in the network. Other policies, for example, energy consumption and sustainability simply do not depend on the network. In addition to meeting the policy, there would still be savings in energy consumption.

REFERENCES

ALALWAN, N.; ZEDAN, H.; SIEWE, F. ***Generating OWL ontology for database integration***. In: 3rd International Conference on Advances in Semantic Processing - SEMAPRO 2009. [S.l.: s.n.] p. 22–31, 2009.

ALMAGOR, R. C. ***Internet history. Journal of technoethics***, v. 2, n. June, p. 45–64, 2011.

ALMEIDA, J. P. A. ***Model-Driven Design of Distributed Applications***. In: CTIT PH.D.-THESIS SERIES, NO. 06-85, Telematica Instituut Fundamental Research Series. Enschede: [s.n.], p. 18, 2006.

BAJWA, I. A. ***Framework for Ontology Creation and Management for Semantic Web***. International Journal of Innovation Management, v. 2, n. 2, p. 116–118, 2011.

BAKER, P. G. et al. ***An ontology for bioinformatics applications***. Bioinformatics, v. 15, n. 6, p. 510–520, 1999. ISSN 13674803.

BALDAUF, M.; DUSTDAR, S.; ROSENBERG, F. ***A survey on context-aware systems***. International Journal of Ad Hoc and Ubiquitous Computing, v. 2, n. 4, p. 263, 2007.

BARCELLOS, M. P.; De Almeida Falbo, R. ***Using a foundational ontology for reengineering a Software Enterprise Ontology***. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). [S.l.: s.n.] v. 5833 LNCS, p. 179–188, 2009.

BASS, L.; CLEMENTS, P.; KAZMAN, R. ***Software Architecture in Practice***. 2. ed. [S.l.]: Addison-Wesley Professional, 2003.

BATEMAN, J. a. et al. ***A linguistic ontology of space for natural language processing***. Artificial Intelligence, v. 174, n. 14, p. 1027–1071, 2010.

BERGAMASCHI, S. et al. ***Semantic integration of heterogeneous information sources***. Data Knowledge Engineering, v. 36, n. 3, p. 215–249, 2001.

BISBY, F. A. ***The quiet revolution: biodiversity informatics and the internet***. Science, v. 289, n. September, p. 2309–2312, 2000.

BLAIR, G. S. et al. ***The role of ontologies in emergent middleware***: Supporting interoperability in complex distributed systems. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). [S.l.: s.n.], 2011.

BROWN, P. ***The Internet Revolution***. [S.l.: s.n.]. v. 24, 2006.

BUNGE, M. Ontology I: ***The Furniture of the World***. In: Treatise on Basic Philosophy. [S.l.]: Cambridge, Mass., Harvard University Press.

CARVALHO, O. de. ***Aristóteles em Nova Perspectiva***. 2. ed. [S.l.]: Vide Editorial, 2013.

CHEN, Y. J.; CHEN, Y. M.; CHU, H. C. ***Development of a mechanism for ontology-based product lifecycle knowledge integration***. Expert Systems with Applications, v. 36, n. 2 PART 2, p. 2759–2779, 2009.

CHEN, Y. J.; CHEN, Y. M.; SU, Y. S. ***An ontology-based distributed case-based reasoning for virtual enterprises***. In: Proceedings of the International Conference on Complex, Intelligent and Software Intensive Systems, CISIS, 2009.

CHOI, N.; SONG, I.-Y.; HAN, H. ***A Survey on Ontology Mapping***. Namyoun Choi, Il-Yeol Song, and Hyoil Han College of Information Science and Technology Drexel University, Philadelphia, PA 19014. SIGMOD Record, v. 35, n. 3, p. 34–41, 2006.

CIULLO, D.; MELLIA, M.; MEO, M. ***Traditional IP measurements: What changes in a today multimedia IP network***. In: Proceedings of the 2008 4th International Telecommunication Networking Workshop on QoS in Multiservice IP Networks, IT-NEWS. [S.l.: s.n.]. p. 262–267, 2008.

CORAZZON, R. ***Notes on the history of ontology, descriptive and formal ontology***. 2013.

CORCHO, O.; FERNANDEZ-LOPEZ, M.; GOMEZ-PEREZ, A. ***Ontological engineering: Principles, methods, tools and languages***. In: Ontologies for Software Engineering and Software Technology. Berlin, Alemania: Springer-Verlag, 2006. p. 1–48.

FARINELLI, F.; ALMEIDA, M. ***Interoperabilidade semântica em sistemas de informação de saúde por meio de ontologias formais e informais***. Conferência Internacional Acesso

Aberto, Preservação Digital, Interoperabilidade, Visibilidade e Dados Científicos, Universidade Federal de Minas Gerais, 2013.

FARQUHAR, A.; FIKES, R.; RICE, J. ***The Ontolingua Server: a tool for collaborative ontology construction***. International Journal of Human-Computer Studies, v. 46, n. 6, p. 707–727, 1997.

FIKES, R.; FARQUHAR, A. ***Distributed repositories of highly expressive reusable ontologies***. IEEE Intelligent Systems and their Applications, v. 14, p. 1–14, 1999.

FIKES, R.; FARQUHAR, a. Distributed repositories of highly expressive reusable ontologies.

FOROUZAN, B. A. ***TCP/IP protocol suite***. McGraw-Hill, 2008.

GOMEZ-PREREZ, A. ***Ontology Evaluation. Handbook on Ontologies***, p. 293–313, 2004.

GROVES, C. et al. ***Gateway Control Protocol Version 1***. IETF. RFC 3525 (Historic). (Request for Comments, 3525), 2003.

GRUBER, T. R. ***A translation approach to portable ontology specifications***. Knowledge Acquisition , v. 5, n. 2, p. 199–220, 1993.

GUIZZARDI, G. ***Foundations for Structural Conceptual. Centre for Telematics and Information Technology***, University of Twente v. 015. 441 p. (Telematica Instituut Fundamental Research Series, CTIT Ph.D.-thesis series No. 05-74), 2005.

HACHEM, S.; TEIXEIRA, T.; ISSARNY, V. ***Ontologies for the internet of things***. Proceedings of the 8th Middleware Doctoral Symposium on - MDS ’11, n. June 2009, p. 1–6, 2011.

HARTMANN, J.; PALMA, R.; GÓMEZ-PÉREZ, A. ***Ontology Repositories***. In: Handbook on Ontologies. [s.n.]. p. 551–571, 2009.

KIONG, Y. C.; PALANIAPPAN, S.; YAHAYA, N. A. ***Health ontology system***. In: 2011 7th International Conference on Information Technology in Asia: Emerging Convergences and Singularity of Forms - Proceedings of CITA’11. [S.l.: s.n.], 2011.

LEINER, B. M. et al. ***A Brief History of the Internet***. ACM SIGCOMM Computer Communication Review , v. 39, n. 5, p. 22–31, 2009.

MEALY, G. H. ***Another Look at Data***. In: Proceedings of the November 14-16, 1967, Fall Joint Computer Conference. New York, NY, USA: ACM, 1967. (AFIPS ’67 (Fall)), p. 525–534, 1967.

NECHES, R. et al. ***Enabling technology for knowledge sharing***. AI Mag. American Association for Artificial Intelligence, Menlo Park, CA, USA, v. 12, n. 3, p. 36–56, set. 1991.

NOY, N. F.; MCGUINNESS, D. L. ***Ontology Development***: A Guide to Creating Your First Ontology, 2011.

NYGREN, E. et al. The Akamai Network: ***A platform for high-performance Internet applications***. ACM SIGOPS Operating Systems Review , v. 44, n. 3, p. 2–19, 2010.

OLIVEIRA SILVA, F. ***Endereçamento por Título: Uma Forma de Encaminhamento Multicast para a Próxima Geração de Redes de Computadores***. Tese (Doutorado) — Universidade de São Paulo, Escola Politécnica, São Paulo, Brasil, 2013.

PARALIC, J.; KOSTIAL, I. ***Ontology-based Information Retrieval***. Information and Intelligent Systems, Croatia, p. 5, 2003.

PEIRCE, C. S. ***Collected Papers of Charles Sanders Peirce***. [S.l.]: Cambridge, Mass., Harvard University Press, 1960.

PENROSE, R. ***The Large, the Small and the Human Mind***. [S.l.]: Cambridge University Press; Reprint edition, 2000.

PEREIRA, J. H. d. S. et al. ***Title Model Ontology for Future Internet Networks***. In: Future Internet Assembly 2011: Achievements and Technological Promises. Future Internet: Achievements and Promising Technology: Springer-Verlag (LNCS, v. 6656). p. 465, 2011.

PEREIRA, J. H. D. S.; KOFUJI, S. T.; ROSA, P. F. ***Distributed systems ontology***. In: 3rd International Conference on New Technologies, Mobility and Security, NTMS 2009.

PEREIRA, J. H. S. et al. ***Layers Optimization Proposal in a Post-IP Network***. International Journal On Advances in Networks and Services , 2011.

PEREIRA, J. H. S. ***Modelo de Título para a Próxima Geração de Internet***. Tese (Doutorado) — Universidade de São Paulo, Escola Politécnica, São Paulo, Brasil, mar 2012.

PEREIRA, J. H. S.; KOFUJI, S. T.; ROSA, P. F. ***Horizontal Address Ontology in Internet Architecture***. In: New Technologies, Mobility and Security ({NTMS)}, 2009.

PETERSON, K. R. ***An Introduction to Nicolai Hartmann's Critical Ontology***. Axiomathes, v. 22, n. 3, p. 291–314, 2012.

POSTEL, J. ***DoD standard Transmission Control Protocol***. IETF. RFC 761. (Request for Comments, 761), 1980.

POSTEL, J. ***Simple Mail Transfer Protocol***. IETF, 1982. RFC 821 (INTERNET STANDARD). (Request for Comments, 821), 1982.

POSTEL, J. ***User Datagram Protocol***. IETF. RFC 768 (INTERNET STANDARD). (Request for Comments, 768), 1980.

POSTEL, J.; REYNOLDS, J. ***File Transfer Protocol***. IETF. RFC 959 (INTERNET STANDARD). (Request for Comments, 959), 1985.

ROSENBERG, J. et al. ***SIP: Session Initiation Protocol***. IETF. RFC 3261 (Proposed Standard). (Request for Comments, 3261). 2002.

SANTUCCI, G. ***The Internet of Things: Between the Revolution of the Internet and the Metamorphosis of Objects***. Forum American Bar Association , p. 1–23, 2010.

SHERRY, J. L.; BOWMAN, N. D. ***History of the Internet***. In: Handbook of Computer Networks . [S.l.: s.n.], 2011. v. 2, p. 280–293. ISBN 9780471784593, 2011.

SMITH, B. ***Ontology. Blackwell Guide to the Philosophy of Computing and Information***, n. 1964, p. 155–166, 2003. ISSN 1943-4723, 1964.

SMITH, D. W. ***Husserl***. [S.l.: s.n.], 2007. 467 p. ISSN 1098-6596. ISBN 0415289742, 2007.

STUDER, R.; BENJAMINS, V. R.; FENSEL, D. ***Knowledge engineering: Principles and methods***. Data Knowl. Eng. Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, v. 25, n. 1-2, p. 161–197, 1998.

SU, X. L. et al. ***Review on the Work of Agriculture Ontology Research Group***. P. 720–730, 2012.

SUN, Y.-C. ***The development of an artificially intuitive reasoner***. 7513 p, , 2011.

TRONCO, T. R. ***A brief history of the internet. Studies in Computational Intelligence***, v. 297, p. 1–11, 2010.

UNION, I. T. ***Security architecture for Open Systems Interconnection for CCITT applications. Recommendation X.800***. International Telecommunication Union (Series X: Data Networks and Open System Communication), 1991.

USCHOLD, M.; GRUNINGER, M. ***Ontologies: Principles, Methods and Applications***. Knowledge Engineering Review, v. 11, n. 02, p. 63, 1996.

VAS, R. ***Educational ontology and knowledge testing***. Proceedings of the European Conference on Knowledge Management, ECKM, p. 577–584, 2006.

WALDROP, M. ***DARPA and the internet revolution***. DARPA: 50 Years of Bridging the Gap, n. December 1969, p. 78–85, 2008.

WEILONG, L.; FANG, J.; XIN, Z. ***Ontology-based user modeling for E-commerce system***. In: 2008 3rd International Conference on Pervasive Computing and Applications, ICPCA08, v. 1, p. 260–263, 2008.

XU, B.; WU, J.; CAI, H. ***Business process driven ontology discovery method from distributed data environment***. In: Proceedings - 2011 8th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2011. v. 2, p. 1246–1251, 2011.

YANG, L. et al. ***Forwarding and Control Element Separation*** (ForCES) Framework. IETF, 2004. RFC 3746 (Informational). (Request for Comments, 3746), 2004. ;