

On Intelligent, Autonomous and Collaborative Agents to Manage Internet Routing Domains

Juliao Braga^{1,3}, Joao Nuno Silva¹, Patricia Takako Endo², Nizam Omar³

¹ INESC-ID, Rua Alves Redol, 9, 1000-029 Lisboa, Portugal

²Universidade de Pernambuco, Recife, PE, Brazil

³Mackenzie Presbyterian University, Sao Paulo, SP, Brazil

{juliao.braga, j.n.silva}@tecnico.ulisboa.pt

patricia.endo@upe.br, nizam.omar@mackenzie.br

Abstract. *This article describes an environment for knowledge acquisition, learning, use and collaboration inter agents over Internet Infrastructure. Four agent types are used in a previously applied four-tier model, such as the use case on the Internet Routing Registry. This model, which can be implemented in each Autonomous System domain of the Internet infrastructure, is integrated into an environment with (a) capturing information from unstructured databases, (b) creating and updating training bases appropriate to machine learning algorithms and (c) creation and feeding of a knowledge base. Such resources become readily available to agents in each domain and to agents in all other domains with the aim of making them autonomous. The agents collaborate and interact with each other, through individual blockchain structures that also take care of operational security and integration aspects. In addition, a test bed to validate the entire model, including the functionalities of the agents, is also proposed and characterized.*

Resumo. *Este artigo descreve um ambiente para aquisição de conhecimento, aprendizado, uso e colaboração entre agentes sobre a Infraestrutura da Internet. Quatro tipos de agentes são usados, em um modelo de quatro camadas aplicado anteriormente, como caso de uso, no Internet Routing Registry. Esse modelo, que pode ser implementado em cada Sistema Autônomo da infraestrutura da Internet é integrado a um ambiente com (a) captura de informações de bancos de dados não estruturados, (b) criação e atualização de bases de treinamento apropriadas a algoritmos de aprendizado de máquina e (c) criação e alimentação de uma base de conhecimento. Tais recursos tornam-se prontamente disponíveis para agentes em cada domínio e para agentes em todos os outros domínios com o objetivo de torná-los autônomos. Os agentes colaboram e interagem uns com os outros, através de estruturas de blockchain individuais, que também cuidam dos aspectos de segurança operacional e integração. Além disso, uma cama de testes para validar todo o modelo, incluindo as funcionalidades dos agentes, também é proposta e caracterizada.*

1. Introduction

This work aims to convey enough intelligence to *Autonomous Architecture Over Restricted Domains* (A2RD) agents so as to make them autonomous. This requires an or-

ganized integration of the resources shown in Figure 1. The A2RD model, which can be implemented in each AS, is represented as item (1) in the figure.

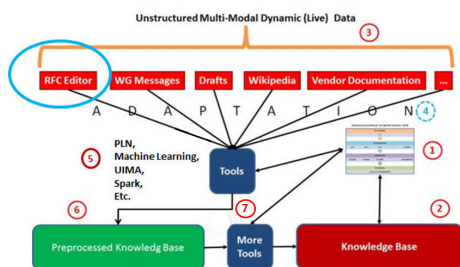


Figure 1. Components that should be operationally integrated and functionally orchestrated to meet the goal

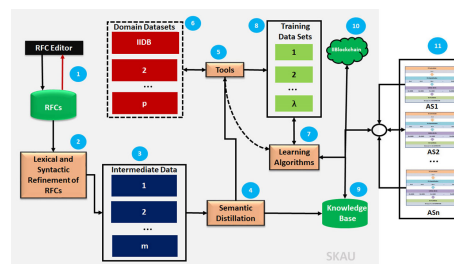


Figure 2. Structure for Knowledge Acquisition, Use, Learning and Collaboration model (SKAU)

A2RD *Intelligent Elements* (IEs) require a *Knowledge Base* (KB) (2) so that they can exercise their autonomy. This KB is dynamically developed from unstructured databases, i.e., documents (3). Knowledge will be acquired through adaptation (4) of captured data. Such adapted data serve as input to special tools available or to be developed (5). These tools create intermediary databases (6) which ultimately feed the KB. Additional tools (7) are shared by the A2RD IEs (1) and when necessary, used to refine the KB and as learning resources for the IEs.

An in-depth analysis of the components framework (Figure 1) allows the development of an A2RD Environment Conceptual Model named *Structure for Knowledge Acquisition, Use, Learning and Collaboration* (SKAU) model (Figure 2). Therein, each implementation of A2RD into an AS is represented as an agglomeration of IEs in a four layer model (11). The SKAU components, which will be described next in the sequence, are dynamically constructed from non-structured databases. The experimental character of this work led to the choice of a subset of non-structured documents stored in the *Request for Comments* (RFCs) repository. This repository contains documents authored by network operators, engineers and computer scientists, documentary methods, behaviors, research, or innovations applicable to the Internet; all resulting from IETF and IRTF working groups and maintained by the RFC-Editor¹. For the focused purpose of the research, subsets of RFCs will be selected with the RFC editor (blue circle Figure 1) such as IRR, DNS, MPLS, OSPF, respecting a specific array of protocols or techniques .

The theoretical SKAU components and activities are:

- RFCs are captured/updated and stored locally (1);
- A set of tools is responsible for acting lexically and syntactically on RFCs (2), transforming them into intermediary databases (3);
- Other tools (4), like *Semantic Distillation*, act on the intermediary databases producing inputs for the construction of *Domain Data Sets* (DDS) (6) and these are built into *Training Data Sets* (TDS) (8). Also, these tools will support part of the Knowledge Base (KB) (9) [Isotani and Bittencourt 2015]. A relevant component

¹<https://www.rfc-editor.org>

of DDS is the Internet Infrastructure Data Base (IIDB), summarized in subsection 1.1;

- *Learning algorithms* (7) support the construction and use of TDS to renew the knowledge base and meet the demand A2RD agents in the process of developing and applied intelligent actions. The efficient use of TDS will respond to the classic algorithms of Machine Learning (ML): (a) supervised learning, (b) unsupervised learning, (c) reinforcement learning and (d) semi-supervised learning, that combines (a) and (b). [Musumeci et al. 2018] in Section II has an appropriate ML overview, with focus on optical networks.
- A database, named *IIBlockchain* (10) is built by each implemented A2RD model and stored together in the Git Hub (i.e. in cloud). This then support the process of collaboration and effective interaction, inter/intra agents of the models [Braga et al. 2018]. The *IIBlockchain* cloud interacts with the *learning algorithm* and KB allowing agents to exercise *offline and online computation*². A brief description of IIBlockchain is found in subsection 1.2.

Each AS can implement an A2RD, which is then controlled by the IE – *IE Controller* – and receives the identification $x:0$, where x is the *AS Number* (ASN).

1.1. IIDB

IIDB evolved from the efforts to build *WordNet* [Fellbaum 1998]. It was later realized that its usefulness would be amplified if it represented not only words (from the domain of the Internet Infrastructure) and their lexical equivalents but also any representation (proper names, numbers, dates, acronyms, etc.) associated with its meaning. Table 1 shows an example of the IIDB contents (excluding implementation details).

Table 1. Partial contents of IIDB

Representation	Meaning	Ext	Sub
1943-08-06:1998-10-16	Jon Postel	img:lnk;text:url	human
IETF	Internet Engineering Task Force	-	ietf
3978	RFC	-	doc
3978	OpenTTD game (masterserver and content service)	TCP;UDP	tcp
S. Bradner	RFC3978	Author	rfc

Approaching the formal notations and definitions of WordNet [Miller 1995], the IIDB is defined as $W_I = (f, s, e, d)$ where f is a *form* composed by a string over a finite alphabet, s is *sense* got from a given set of meanings found in the unstructured bases shown in (3) from Figure 1, e is an *extension* which complements s and d is the sub-domain to which *form* s belongs.

Hence, IIDB is a data set that covers the Internet Infrastructure domain and can be used for quick access not only to IE but also by third parties and will help build the KB and support the IEs in updating this KB and will be used as learning content for ML algorithms. The first three databases (iidb.rfc: 3.8GBytes, iidb.person: 4.9GBytes and iidb.acronym: 9.4GBytes) are available in Open System Foundation³ (OSF) repository [Braga et al. 2019].

²*Offline computation is the computation done by the agent before it has to act, and online computation is the computation done by the agent between observing the environment and acting in the environment* [Poole and Mackworth 2010]

³<https://osf.io/nzfqw/>

1.2. IIBlockchain

IEs need to communicate in order to collaborate, learn and cooperate with each other. This communication requires security, i.e., the respective IE controller must recognize the origin of each pair in their information exchanges. *Pretty Good Privacy* (PGP) was found to be a simpler alternative to the IE controller [Garfinkel 1995]. Using PGP, an AS_x IE controller wishing to communicate with an AS_y IE controller, will use the AS_y *public key* to encrypt the message, for $\forall x$ and $\forall y$ such that $x \neq y$ and $x, y = 1, \dots, n$, $n \leq$ total ASes present in the *Internet Routing Table*⁴. The AS_y controller uses its *secret key* to decrypt the message. Thus, for this and for other reasons identified in the following section, the recommended solution was a variation of blockchain implementations proposed in the literature, here named as *IIBlockchain* [Marwala and Xing 2018] [Xu et al. 2016] [Witte 2016] [Prusty 2017].

2. State of the Art

2.1. Theoretical Foundations

Intelligent agents [Russel and Norvig 2010], have been considered viable to be applied to the various areas of knowledge associated with the resources and facilities that make the Internet work. As stated above, the interests were to apply intelligent techniques over agents, called IE, in restricted domains of the Internet Infrastructure, that is, in the so-called routing domains represented by AS [Braga and Omar 2014].

Many terms used or need to be used to form the set of concepts necessary to apply the specific notion of IEs are still not well defined or clear in the literature. To avoid ambiguities and therefore to leave the understanding clear, it is necessary to present some definitions that will be used in this article.

2.1.1. Self-organization

The term *self-organization* was defined in the early 1970s Ilya Prigogine, Nobel Prize in Chemistry in 1977 [Ebeling and Feistel 2011]. In the context of this work and when dealing with intelligent agents, *self-organization* is meant the ability of an agent to readily react when it perceives a threat (or instability) in some way, hence indicating the possibility of a deviation in its functional objectives. This reaction returns the functionality of the agent to its stable condition in the previous state (self-organized) to the extraordinary event of the environment (Simplified definition in Figure 3).

An **intelligent agent** have certain properties and appropriate functional characteristics to be able to self-organize (discussed later in detail). In the context of this research, if an intelligent system exercises its capacity for self-organization without absolutely no human intervention it will be recognized as **autonomous**, with definite freedom to act [Agoulmine 2010]. However, if the intelligent system depends on a no-direct human orientation to exercise the function of self-organization, i.e., on pre-defined parameters indicating how to react, then the given denomination will be **autonomic**. In order words, an **intelligent system** is **autonomic** when reacting to an action of the environment if there is a Human orientation (a plan) on how it should behave to self-organize, when reacting

⁴<http://thyme.rand.apnic.net/current/data-summary>

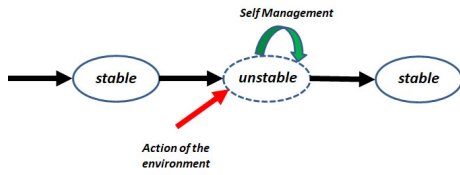


Figure 3. Self-organization. Adapted from [Ebeling and Feistel 2011]



Figure 4. Characterization of intelligence and independence of Intelligent Systems in the A2RD project

to an action of the environment. If an intelligent system performs functions integrally inserted by a Human in order to exercise its capacity to self-organize, then it is considered **automatic**.

Finally, if the intelligent system is unable to self-organize under an unusual action of the environment, it eventually fails, then it is said to be **legacy**. The relationship between such intelligent systems, indicating the degrees of independence, intelligence aggregates and how it will be interpreted in this work is shown in Figure 4. The interpretation of **independence** is linked to human participation. The more intense that participation, the less autonomous the system is. The notion of **intelligence** is associated with the ability of **self-organization** of the system. The greater the ability to effectively use the properties and functionality of self-organization, the more autonomous the system is. On the other hand, the intelligence and independence terms are directly associated with the learning ability of the intelligent elements. This learning will be effective if there is **cooperation** between the intelligent elements, without distinction. The project is not interested in the functional aspects of **legacy systems** but, whenever necessary, will use results or configuration parameters of these systems, to maintain balance between all the elements involved.

2.1.2. Domains

A *domain* represents a collection of things (actors, entities, etc.) that are aligned and united through common goals, within the specific limits of an area of interest⁵. In the context of the A2RD project, the area of interest is the environment outlined by the activities associated with the ASs which together represent the Internet. In turn, ASs have subsets of interests identified as *sub-domains*. In general, the environment where *Internet of Things* (IoT) techniques are applied is a real example of a sub-domain.

2.1.3. Interoperability and Ontology

Interoperability is the term used in this context to designate the ability to cooperate between domains and/or sub-domains through its respective IEs, for the achievement of common objectives. There are two types of interoperability: *syntactic interoperability*

⁵<http://www.ncoic.org/what-is-interoperability>

and *semantic interoperability*. *Syntactic interoperability* refers to information that is exchanged between IEs during the connection. In general, this information is inserted in the context of the protocols enabling the connectivity. Eventually, it uses additional information, which ascends to the upper layers of the TCP/IP model, to maintain the connection that does not need interpretation of meaning or is just pure data. In this case, some features such as *eXtensible Markup Language* (XML) or *Structured Query Language* (SQL) enable interoperability in formats recommended and described in this document.

After the interconnection the IEs must maintain their communication and cooperation process. This represents *semantic interoperability*. The exchange of information between IEs begins to produce data with understandable meaning, whose interpretation is appropriate to produce the expected results. In appropriate repositories such data are accompanied by a special formatting called *ontology*. Ontology is the resource used to represent knowledge. The ontology – the knowledge associated with data in the pure state – has adequate languages to serve it. These languages vary depending on its ability to clearly express desired knowledge. Although details are discussed later in this text, the Figure 5 displays the main languages available in a comparison of its formalism and its ability to express knowledge. Note that the most powerful language is the natural language, but this will not be used in this work because it restricts interpretation by non-Human procedures.

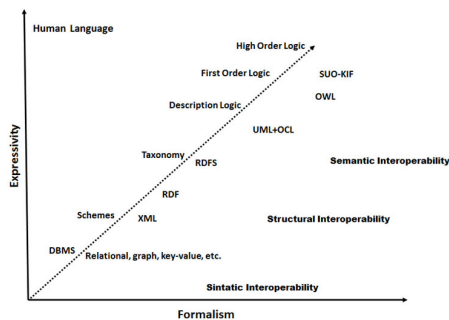


Figure 5. Languages to represent knowledge and its expressivity. Source: [Peace 2011]

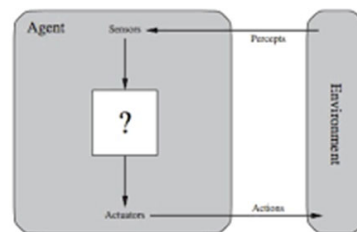


Figure 6. How the agents interact with the environment. Source: [Russel and Norvig 2010]

2.1.4. Intelligent Agents

Based on his doctoral thesis of 1992, Michael Wooldridge made a considerable effort in the formalization of intelligent agents [Wooldridge 1992]. He also established the theoretical and practical concepts involving intelligent agents. Its architecture, mathematical models, logic and semantics, as well as considerations about the issues surrounding software projects and development, with approaches on the multi-disciplinarity around of intelligent agents [Wooldridge and Jennings 1995]. This same work recalls that a cluster of intelligent agents form a **agency**. An agency consists of *multi-agents*. On two occasions, Wooldridge consolidated the ideas on *multi-agents* [Wooldridge 2002] [Wooldridge 2009]. In the second edition, using an article he defines, in free translation [Wooldridge and Jennings 1995]:

“An agent is a computer system that is situated in some environment, and that is capable of autonomous action in this environment in order to meet its delegated objectives” [Wooldridge 2009, Chapter 2].

On the other hand, Russel and Norvig [Russel and Norvig 2010], presented a simplified concept of the relationship between the agent and the environment in which it is associated (Figure 6). According to them, *an agent is something that has perception of its environment through sensors and acts on the environment through actuators*.

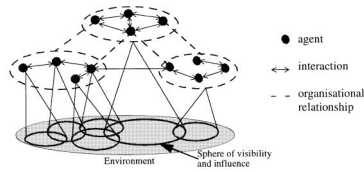


Figure 7. Canonical view of an agent-based system. Source: [Jennings 2000, page 281]

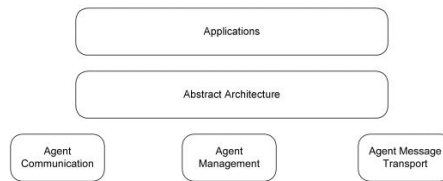


Figure 8. FIPA specification category groups

[Russel and Norvig 2010, Wooldridge 2009, Wooldridge 2002] are appropriate texts for the learning and application of agents and multi-agents. The first, with its broad description of Artificial Intelligence techniques and the second and the third, specific texts about multi-agents. To them, join a book, also, essential and complementary [Wooldridge 2000]. In addition, one can not forget [Weiss 1999]. Everyone at the end of each chapter makes appropriate and convincing approaches to the issues discussed above, including a literature review. [Russel and Norvig 2010] present a more up-to-date source.

2.1.5. Multi-Agents

In addition to the definitions of agents given in the previous sections, a very clear multi-agent model proposed by [Jennings 2000] is represented in Figure 7. This model is suitable to establish the main motivation of the proposal of the A2RD model. Before proceeding with this analysis of the peculiarities and specific properties of an IE, it is worth noting the definition given by the same author:

“An agent is an encapsulated computer system that is situated in some environment and that is capable of flexible, autonomous action in that environment in order to meet its design objectives.”

Immediately, the same author clarifies some points associated with the definition of his canonical model. According to [Jennings 2000], agents are:

- (a) *clearly identifiable*: as entities that solve problems, with well-defined interfaces and boundaries;
- (b) *encapsulated in a particular environment*: they receive through sensors inputs related to the state of its environment and act on the environment through actuators (which he calls *effectors*);
- (c) *designed to meet a specific goal*: they have particular goals to meet;

- (d) *autonomous*: they have control over both its internal state and its own behavior. The control characteristic about its self-behavior is what distinguishes agents from objects;
- (e) *able to exhibit flexible problem solver behavior*: in addressing its goals they need to be both reactive (able to respond in time to changes that occur in its environment) and proactive (empowered to act ahead of the future goals).

These observations, complemented by [Wooldridge and Jennings 1995] [Wooldridge 1997] highlight that multi-agents definitions is a very active and on-going area of research. Two aspects stand out in the analysis of the Figure 7:

- (i) From the point of view of interconnection a complete mesh or full mesh is absent, i. e., some agents do not communicate directly with other agents.
- (ii) In relation to the environment, the agents act in specific sub-domains and more than one agent can act on the same sub-domain.

2.1.6. Communication between Intelligent Agents

There is a prominent research effort in the direction of defining the communication properties between agents. One such initiative is the *Foundation for Intelligent Physical Agents*⁶ (FIPA). founded in 1995 with the aim of developing standards aimed at systems of agents. These standards, are grouped⁷ in specified categories according to Figure 8. FIPA standards systematically use Software Engineering and Unified Modeling Language (UML), as meta-language [Eriksson and Penker 2004] [Eriksson and Penker 2000]. In fact, when dealing with agents, FIPA recommends extensions for UML. In a search of common aspects, [Bernon et al. 2005] present a comparison of several meta-models used in some known methodologies. In addition, in a UML approach, [Bauer et al. 2001] provides a very expressive overview and presents an extension of the UML, called Agent Unified Modeling Language (AUML). [Odell et al. 2001] describe the UML meeting the specification [FIPA 2002b]. In [Bauer et al. 2000], the AUML extension is described comprehensively. Object Management Group (OMG) discusses agents⁸, and recommends a specific environment for AUML⁹. The Abstract Architecture [FIPA 2002a] is the prerequisite for progressing to other specifications. In [FIPA 2004], the proposal for an inter-agent communication language (ACL) that gave rise to Java Agent Development Framework (JADE), whose best-known original document is [Bellifemine et al. 1999] followed by a complementary article [Bellifemine et al. 2008] and a much more complete text in [Bellifemine et al. 2007]. The importance of the environment, in which the agents interact is characterized in a very precise approach in [Odell et al. 2003].

Whether it is an attempt to standardize or not, the FIPA proposal is not the only one. For example, in cases of agents in *peer-to-peer* environments, described in [Moro et al. 2005]. This project deviates considerably from FIPA's proposal, making the desired scenario for the IEs Agency more flexible and closer to the characteristics of the Internet Infrastructure when necessary.

⁶<http://www.fipa.org>

⁷<http://www.fipa.org/repository/bysubject.html>

⁸<http://www.objs.com/agent/>

⁹<http://www.auml.org/>

2.2. Related Work

Table 2 identifies the main works which strongly influenced the development of A2RD and their respective characteristics involving intelligent agents.

Table 2. Related works

Characteristics	MAPE-k	Others	Schmid	ANIMA
Reference	IBM [Horn 2001]	[Movahedi et al. 2012]	[Schmid et al. 2006]	[Behringer et al. 2014] (IRTF + IETF)
Domain	Application. IBM Products.	Application. Autonomic Architecture	Networks	Networks: autonomic nodes with the same intention
Integration between elements	Through an executor	Follow the MAPE-K: executor equivalent	Variation of MAPE-K	If necessary, use the Feedback Cycle
ID	Undefined	Undefined	Undefined	IPv6 (host interface)
Human Interference	High Level Objective	High Level Objective	High Level Objective. Deterministic Behavior	Intention. Autonomic Control Plane
Specific	Self management	Self management	Self management and self-adaptation	Self-management, network knowledge, self knowledge (self-awareness)
Architecture	Centralized, restricted scalability	Hierarchical, peer, restricted scalability	Similar to FIPA's proposal: Agency	Scalable in the domain

The associated ideas originated from the proposal of Autonomic Computing [Horn 2001]. [Movahedi et al. 2012] display details of MAPE-K model with its respective control cycles and compare several other proposals of autonomic architectures with emphasis on networks. [Schmid et al. 2006] proposes changes in the MAPE-K model, simplifying it, for elements of autonomic networks. [Behringer et al. 2014] started in the Internet Research Task Force (IRTF) NMRG group, proposals that continued in the IETF ANIMA group, with studies that are in full activity, with specific recommendations for protocols, indicating the most recent and active studies on autonomic networks. From the models described in Table 2, only ANIMA proposes an identification (ID) for its autonomic functions, associating them with an IPv6 address, indicating that the autonomic functions are aggregated to the host interface.

Contrary to the literature and considering Section 1, the present work considers timely to improve the intelligence of the agents that inhabit the A2RD model, making them autonomous and truly independent of the ASs administrators.

3. Methodology

We designed a testbed to validate the theoretical model, as shown in Figure 9.

It involves three environments and a set of appropriate resources to test all the requirements of the proposed model, specifically:

- (i) One AS with a public ASN, especially obtained for the test-bed, AS204466¹⁰, where to hosted one independent A2RD;
- (ii) One public IPv6 block announced on the Internet: 2a05:7541:4000::/34;
- (iii) Two local environment with private ASN, where two independent A2RDs will be hosted;
- (iv) Private IPv4 in all environments;
- (v) Python language for all required code and if necessary the language C or C++;
- (vi) Availability of the code and data in the OSF¹¹;
- (vii) Use of the domain *a2rd.pt* and DNS servers (one in each ASN and one of them an hidden master);

¹⁰<https://bgp.he.net/AS204466>

¹¹<https://osf.io/tka9u/>, DOI 10.17605/OSF.IO/TKA9U

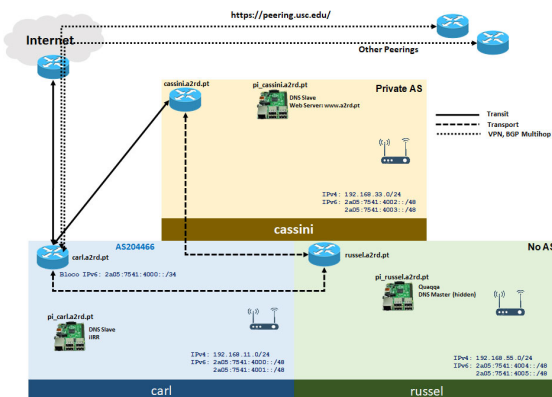


Figure 9. A2RD and SKAU testbed

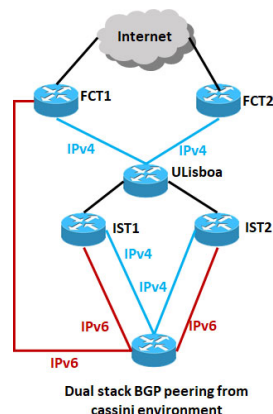


Figure 10. Test-bed external topology, with the respective devices and dual stack BGP peer from cassini environment

(viii) Use of the Zope¹² for the web server of the project and as hosting environment for the code will be developed.

4. Expected Outcome

This research project is expected to provide a conceptual proof that the IEs act as autonomous agents, learning from a subset of non-structured documents in order to build (and renew) knowledge. To achieve this, the works defines, specifies and characterizes the IEs required to meet these requirements.

We also provide a specification of an IEs development methodology, similar to INTERA, a learning object development methodology [Braga 2015]. It is expected that the knowledge captured from unstructured databases have positive results and that their algorithms are identified and qualified. On the other hand, we also define the KB storage format, structure, mode and location with precise and practical fundamentals from the point of view of its usability by the IEs.

We will conduct an evaluation with some ML algorithms, combining learning with the improvement of the knowledge base by the IEs in order to efficiently choose the best ML algorithms available to adopt the recommendation that *Learning = Representation + Evaluation + Optimization* [Domingos 2012].

The experience with the IIDB enables the recognition of an effective utility proposal for other applications on the Internet infrastructure. Thus, for better use of the IIDB and resulting from the efforts of data analysis obtained from unstructured databases, we hope to characterize the structure and organization. Also, the use case - which could be the IRR - is expected to be presented at its completeness effectively without any human intervention.

Finally, we expect to establish that the choice of IIBlockchain can be an effective collaboration technique in the relationship between intelligent agents. Indeed, we aim to

¹²<http://www.zope.org/en/latest/>

show that IIBlockchain is effectively an important collaboration technique between the IEs of the different application domains.

5. Stage of Research

The *dual stack*¹³ external topology of the test-bed was implemented and can be seen in Figure 10, the border connection with the *Fundacao para a Ciencia e a Tecnologia*¹⁴ (FCT) and intermediate BGP peer connections with *Instituto Superior Técnico*¹⁵ (IST) and *Universidade de Lisboa*¹⁶ (ULisboa). Using a *ping*¹⁷ through the IP 193.136.134.244 or 2001:690:2100::aa:2:1 (also via cassini ethernet IPv6: 2a05:7541:4000:0:192:168:11:1).

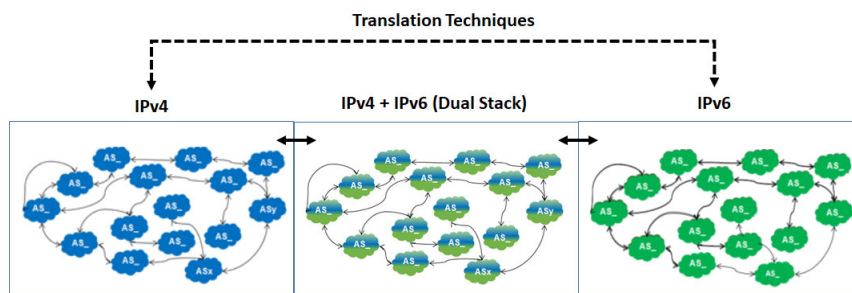


Figure 11. Types of Internet formed by the ASs.

To understand the meaning of a *dual stack* BGP implementation it is necessary to acknowledge the possible existence of three types of AS's agglomerations. Figure 11 shows these three types. On the left you see the ASs that only use the IPv4 protocol, i. e., the *IPv4 stack*, as you commonly say. On the right, you see the ASs that use the *IPv6 stack*, only. Resources (clients and servers) belonging to each AS of these two implementations can only speak to each other, if a translation technique is used, in their BGP connection [Bagnulo et al. 2011] [Wu et al. 2013] [Enache and Alexandru 2016]. However, if, neighbors ASs implement the two protocol stacks in their BGP, then they are able to speak in both IPv4 and IPv6, as shown in the center of the figure [Marques and Dupont 1999]. Due to the exhaustion of IPv4, the test-bed was implanted without this protocol. However, Network Address Translation (NAT) was used, making it possible to implement the two protocol stacks [Srisuresh and Holdrege 1999]. All servers and devices, such as wireless routers, proposed in the Figure 2, are operational. VPNs for external access to all environments are already running. Zope servers have been tested and are operational, awaiting the experiments. IIBlockchain routines are already implemented, and so, ready for use by IEs in testbed. The IIDB is ready with three data bases as described in 1.1, with files totaling 14.7 MBytes, captured data from 7,844 files (RFCs). Base *iidb.rfc* is with 8,508 items, base *iidb.person* is with 5,424 items and base *iidb.acronym* has 11,689 different acronyms (non-repetitive) or about 65,485 repetitive acronyms.

¹³Implementation of BGP with both IPv4 and IPv6 protocols [Martinsen et al. 2018].

¹⁴<https://www.fct.pt/index.phtml.en>

¹⁵<https://tecnico.ulisboa.pt/en/>

¹⁶<https://www.ulisboa.pt/en>

¹⁷Ping is an utility program that allows to verify if a particular IPv4 or IPv6 address is active.

Acknowledgment

This work is supported by CAPES – Brazilian Federal Agency for Support and Evaluation of Graduate Education within the Brazil’s Ministry of Education, by national funds through FCT with reference UID/CEC/50021/2019 and by MackPesquisa.

References

- Agoulmine, N. (2010). Introduction to autonomic concepts applied to future self-managed networks. *Autonomic Network Management Principles: From Concepts to Applications*, page 1.
- Bagnulo, M., Matthews, P., and van Beijnum, I. (April 2011). Stateful nat64: Network address and protocol translation from ipv6 clients to ipv4 servers. Technical report, RFC Editor. RFC6146.
- Bauer, B., Muller, J. P., and Odell, J. (2000). An extension of UML by protocols for multi-agent interaction. In *MultiAgent Systems, 2000. Proceedings. Fourth International Conference on*, pages 207–214. IEEE.
- Bauer, B., Müller, J. P., and Odell, J. (2001). Agent UML: A formalism for specifying multiagent software systems. *International Journal of Software Engineering and Knowledge Engineering*, 11(03):207–230.
- Behringer, M., Pritikin, M., Bjarnason, S., Clemm, A., Carpenter, B., Jiang, S., and Ciavaglia, L. (2014). Autonomic networking - definitions and design goals. Technical report, RFC Editor.
- Bellifemine, F., Caire, G., Pogg, A., and Rimassa, G. (2008). JADE: A software framework for developing multi-agent applications. Lessons learned. *Information and Software Technology*, 50(1):10–21.
- Bellifemine, F., Poggi, A., and Rimassa, G. (1999). JADE–A FIPA-compliant agent framework. In *Proceedings of PAAM*, volume 99, page 33. London.
- Bellifemine, F. L., Caire, G., and Greenwood, D. (2007). *Developing Multi-Agent Systems with JADE*. Wiley, USA, 1 edition.
- Bernon, C., Cossentino, M., and Pavón, J. (2005). Agent-oriented software engineering. *The Knowledge Engineering Review*, 20(02):99–116.
- Braga, J. (2015). *Objetos de Aprendizagem: Metodologia de Desenvolvimento*. Editora da UFABC, São Paulo, 1 edition.
- Braga, J. and Omar, N. (2014). Semantic repository in internet infrastructure knowledge domain: Methodology. In *Anais do CSBC 2014 - IWPIETF LAC*, pages 2179–2184, Brasília, DF.
- Braga, J., Silva, J. N., Endo, P. T., and Omar, N. (2019). Internet Infrastructure Data Base (IIDB). DOI 10.17605/OSF.IO/NZFQW. Available at <https://OSF.IO/NZFQW>. Accessed: 19 Mar 2019.
- Braga, J., Silva, J. N., Endo, P. T., Ribas, J., and Omar, N. (2018). Blockchain to Improve Security, Knowledge and Collaboration Inter-Agent Communication over Restrict Domains of the Internet Infrastructure. In *Proceeding of CSBC 2018 - V Workshop pre IETF*, pages 61–73, Natal, RN Brazil.

- Domingos, P. (2012). A few useful things to know about machine learning. *Communications of the ACM*, 55(10):78–87.
- Ebeling, W. and Feistel, R. (2011). *Physics of Self-organization and Evolution*. John Wiley & Sons.
- Enache, D. and Alexandru, M. (2016). A Study of the Techonology Transition from IPv4 to IPv6 for an ISP. *Review of the Air Force Academy*, 1(31):117–122.
- Eriksson, H.-E. and Penker, M. (2000). *Business Modeling with UML*. OMG Press, New York, USA.
- Eriksson, H.-E. and Penker, M. (2004). *UML 2 Toolkit*. OMG Press, New York, USA.
- Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA.
- FIPA (2002a). Sc000011: Abstract architecture specification.
- FIPA (2002b). Sc00036h: Propose interaction protocol specification.
- FIPA (2004). Sc00023k: Agent management specification.
- Garfinkel, S. (1995). *PGP: pretty good privacy*. O’Reilly Media, Inc.
- Horn, P. (2001). Autonomic computing: Ibm’s perspective on the state of information technology. *IBM Research*.
- Isotani, S. and Bittencourt, I. I. (2015). *Dados abertos conectados*. Novatec Editora, São Paulo, SP, Brasil.
- Jennings, N. R. (2000). On agent-based software engineering. *Artificial intelligence*, 117(2):277–296.
- Marques, P. and Dupont, F. (March 1999). Use of bgp-4 multiprotocol extensions for ipv6 inter-domain routing. Technical report, RFC Editor. RFC2545.
- Martinsen, P., Reddy, T., and Patil, P. (July 2018). Guidelines for Multihomed and IPv4/IPv6 Dual-Stack Interactive Connectivity Establishment (ICE). Technical report, RFC Editor. RFC8421.
- Marwala, T. and Xing, B. (2018). Blockchain and artificial intelligence. *arXiv preprint arXiv:1802.04451*.
- Miller, G. A. (1995). WordNet: A Lexical Database for English. *Communications of the ACM*, 38(11):39–41.
- Moro, G., Sartori, C., and Singh, M. P. (2005). *Agents and Peer-to-Peer Computing*. Springer.
- Movahedi, Z., Ayari, M., Langar, R., and Pujolle, G. (2012). A survey of autonomic network architectures and evaluation criteria. *Communications Surveys & Tutorials*, 14(2):464–490.
- Musumeci, F., Rottondi, C., Nag, A., Macaluso, I., Zibar, D., Ruffini, M., and Tornatore, M. (2018). A Survey on Application of Machine Learning Techniques in Optical Networks. *IEEE Communications Surveys & Tutorials*, pages 1–1.

- Odell, J., Parunak, H. V. D., and Fleischer, M. (2003). Modeling agents and their environment: The communication environment. *Journal of Object Technology*, pages 39–52.
- Odell, J. J., Parunak, H. V. D., and Bauer, B. (2001). Representing agent interaction protocols in UML. In *Agent-oriented software engineering*, pages 121–140. Springer.
- Peace, A. (2011). *Ontology: A Practical Guide*. Articulate Software Press, Angwin, CA, first edition.
- Poole, D. L. and Mackworth, A. K. (2010). *Artificial Intelligence: foundations of computational agents*. Cambridge University Press.
- Prusty, N. (2017). *Building Blockchain Projects*. Packt Publishing Ltd.
- Russel, S. and Norvig, P. (2010). *Artificial Intelligence*. Prentice Hall, New York, 3 edition.
- Schmid, S., Sifalakis, M., and Hutchison, D. (2006). Towards autonomic networks. In *Autonomic Networking*, pages 1–11. Springer.
- Srisuresh, P. and Holdrege, M. (August 1999). IP Network Address Translator (NAT) Terminology and Considerations. Technical report, RFC Editor. RFC2663.
- Weiss, G. (1999). *Multiagent systems: a modern approach to distributed artificial intelligence*. MIT press.
- Witte, J. (2016). The Blockchain: A Gentle Introduction. *SSRN Electronic Journal*.
- Wooldridge, M. (1997). Agent-based software engineering. *IEEE Proceedings-software*, 144(1):26–37.
- Wooldridge, M. (2002). *An introduction to Multiagent Systems*. John Wiley & Sons, 1 edition.
- Wooldridge, M. (2009). *An introduction to multiagent systems*. John Wiley & Sons, 2 edition.
- Wooldridge, M. and Jennings, N. R. (1995). Intelligent agents: Theory and practice. *The knowledge engineering review*, 10(02):115–152.
- Wooldridge, M. J. (1992). *The logical modelling of computational multi-agent systems*. PhD thesis, Citeseer.
- Wooldridge, M. J. (2000). *Reasoning about rational agents*. MIT press.
- Wu, P., Cui, Y., Wu, J., Liu, J., and Metz, C. (2013). Transition from ipv4 to ipv6: A state-of-the-art survey. *IEEE Communications Surveys & Tutorials*, 15:1407–1424.
- Xu, X., Pautasso, C., Zhu, L., Gramoli, V., Ponomarev, A., Tran, A. B., and Chen, S. (2016). The blockchain as a software connector. *Proceedings - 2016 13th Working IEEE/IFIP Conference on Software Architecture, WICSA 2016*, 11(2016):182–191.