Multicast Traffic Aggregation through Entity Title Model

Maurício Amaral Gonçalves, Flávio de Oliveira Silva,
João Henrique de Souza Pereira and Pedro Frosi Rosa
Federal University of Uberlandia
Uberlandia - Minas Gerais - Brazil
Email: mauricioamaralg@gmail.com, flavio@facom.ufu.br,
joaohs@ufu.br, frosi@facom.ufu.br

***Abstract*—Internet was designed in a totally different context than the one existing today. New applications have brought a new set of requirements which were not properly resolved due to architectural limitations. Therefore, the Internet architecture must be reviewed in a *clean slate* approach. In this context, Entity Title Model represents a revolutionary way to semantically understand the entities, observing their needs and capabilities in order to better serve them, through a new flexible architecture with several innovations, especially in addressing and routing aspects. This paper presents a protocol capable of providing efficient multicast at the network layer, based on ETArch over OpenFlow. Multicast is an important requirement for applications involving the transmission of multimedia content, real-time communication and data-sharing services. We describe some experiments and present a comparison between a video application, first implemented using TCP/IP with unicast and multicast services, and then using ETArch focusing on multicast traffic aggregation. The results showed that the bandwidth consumption using our architecture remains constant just as the traditional one; however, our approach uses slightly less bandwidth, provides better strategies for the control plane, improves the group addressability, and facilitates its deployment based on the broad support to Openflow by leading equipment suppliers.**

***Keywords*—telecommunications networks; Internet; multicast; future Internet; clean slate; entity title model.**

I. Introduction

The main concepts of Internet were designed in the sixties [1], and its core protocols were created in the early seventies [2]. If, on one hand, the stability of these protocols led to the the popularity of the Internet, on the other hand, they now refrains its modernization [3]. After four decades and a huge success, much of the initial design of the Internet is still in place. However, applications vastly different from those that initially used the network are now being deployed, bringing a new set of requirements, such as multicast, which current Internet is not able to satisfy in a proper way due to its limitations [4].

Multicast is the ability to deliver data to a group of target entities simultaneously in a single transmission. This aspect is closely linked to how addressing occurs and what routing algorithms are used to reach the entities over the network. The main problem of *Internet Protocol* (IP) addressing is in its ambiguous addressing, which represents both location and identification [5]. This limitation prevents the addressing of a multicast group natively, because there is no unique physical location for a multicast group, and so, the IP address could not be used to locate the members. IP Multicast [6] skirted this problem by using specific reserved address blocks and an implementation of data replication in routers, which became responsible for maintaining the multicast groups. Given the complexity and limitations of this approach, the IP multicast is still not widely used today, even after twenty years of its conception [7].

Researchers from all over the world are engaged in the design of a new Internet from scratch. The *clean slate* approach frees the research from the legacy and fosters innovations [8]. One approach that has taken power in recent years is the *Software-Defined Networking* (SDN), designed in a partnership between UC Berkeley and Stanford University. The Software Defined Networks represents a milestone for advanced researches on new architectures of computer networks. The decoupling between control plane and data plane in network devices contributed with the arising of numerous research projects that collaborated to get the SDN level of maturity as it is today.

Entity Title Architecture (ETArch) presents a vision of how entities are enabled to semantically specify their requirements and capabilities, in order to establish a communication between two or more entities, using a naming scheme based *titles*, which are topology independent and unambiguously designations, and new approaches for addressing and routing aspects [9]. In this work, the ETArch implementation was based on Openflow [10], and focuses on multicast capability, but is not limited to this approach or to this requirement.

The remainder of this paper is organized as follows: Section II describes the related work; Section III presents the Entity Title Architecture; Section IV details the implementation; Section V describes the experiment; Section VI discusses the results obtained and Section VII presents some concluding remarks and potential future works.

II. Related Work

SDN [11][12] represents an extraordinary opportunity to rethink computer networks. It consists of an abstraction that separates the software that controls the network elements from the forwarding plane, providing an open and well-defined interface to control and modify the behavior of network at runtime.

Copyright (c) IARIA, 2014. ISBN: 978-1-61208-360-5

The Future Internet subject is benefited by the range of possibilities offered by the SDN in various applications. In [13], for example, the *Border Gateway Protocol* (BGP) gets an important reinforcement from a *Routing Control Platform* (RCP) system-based, also controlled by SDN. Alternatives to support different applications requirements, such as delivery guarantee, appears in contrast to traditional TCP/IP, as presented by *Dias et al.* [14].

In the state of the art of the SDN, there is an increase in the number of network elements that support OpenFlow; however, although SDN has brought to light the possibility of inferring in the network programming behavior, this is not an easy task. The researchers are engaged in creating software able to abstracting the various features controlled by the network, such as *Foster et al.* [15] and *Kim and Feamster* [16], which offer important contributions for the advancement of researches in this area.

In the EU (European Union), about a hundred different projects are funded under the Seventh Framework Programme (FP7), and some of which are directly related to the Future Internet as 4WARD, CHANGE, MEDIEVAL, PURSUIT, SAIL, SENSEI, TRILOGY and UNIVERSELF [17]. These projects work with different aspects of future networks, and many of them present *clean slate* approaches.

The 4WARD *Netinf* [18] presents an information-centric networking paradigm, based on a distributed system over the network, which controls the communication and provides useful services, such as caching, storage and transporting. It uses a naming scheme independent of the network, called *Identifier*, which is related to *Title* presented at this work. These identifiers are used to register and resolve *Information Objects*, which are primitives exchanged during communication.

In the United States, the *Future Internet Architectures* (FIA) [20], which represents a consolidation of the previous program contains four projects that currently are dealing with aspects of the network, such as content-centric networks, mobility, cloud computing and security. The MobilityFirst [21] network architecture focuses on mobility and propose new protocol stack that considers a new naming scheme based on *Globally Unique Identifier* (GUID) that can provide mobility and multicast. The *Title* is related with the GUID, but the concept of workspace provides a out-of-band control for packet delivery, while in MobilitFirst the control happens in-band.

The IP Multicast, proposed by *Deering* [6], presents limitations both in technical and business aspects [22], such as: limited number of multicast addresses, inability of managing groups dynamically, security constraints, complex architecture, and difficulties in deployment and management.

In IPv6, the concept of broadcast addresses was replaced by the multicast addresses [23]. Furthermore, the network interfaces became able to join different multicast groups. This architecture provides dynamic IP address allocation [24], which can be defined in different scopes [25].

The multicast based on IPv6 presents challenges regarding security [26], with vulnerabilities that can be exploited by attacks. Moreover, scenarios with mobility requirements, where users share frequencies with limited bandwidth, present a number of challenges [27], fueled by the combination of these two requirements.

Due to these limitations, the deployment of IP Multicast occurs slowly [28], which promoted the adoption of the *Application Layer Multicast* (ALM) [29], also known as *End System Multicast* (ESM), in which most of the issues of multicast over IP are addressed at the application layer, facilitating its adoption by not implying changes in the network architecture.

The ability to easily deploying the ALM protocol is a great advantage compared to IP Multicast, which in other hand provides a better optimization of communication bandwidth, partially wasted in ALM due to its multicast strategy, which is based on packet replication over the distribution trees [29]. Moreover, even using ALM, issues such as mobility presents several challenges due to limitations imposed by the architecture.

In this scenario, with different designs, the Entity Title Architecture is an additional proposal that may contribute to this area of research. The outlook presented supports the main ideas about this work, which are: a new protocol stack for the Internet replacing TCP/IP stack, a new naming and addressing scheme, an experimental approach using SDN, an implementation of real multicast, and a vision for collaboration between research community.

III. Entity Title Architecture

ETArch is a *clean slate* approach for the Future Internet, which proposes: a separation of responsibilities between the data and control planes, a semantic proximity of the layers, and a new strategy to addressing and routing. It works as an intermediary layer, as shown in Figure 1. To properly understand how this architecture works, it is first necessary to understand a few concepts:

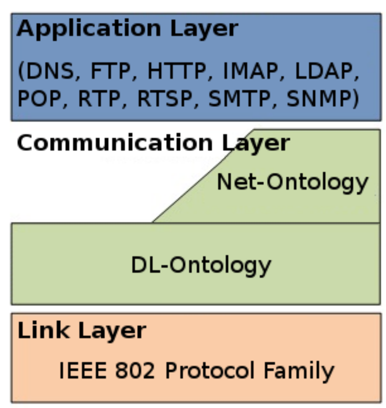

Figure 1. ETArch Stack.

- Entity: is a thing with communication requirements which can be semantically understood from top to bottom layers. Some examples: a content, a service, a sensor device, pad or smart phone, a user, an application, a system, a process. The entity has some titles, requirements and a variable location over time.

- Title: is a designation to ensure an unambiguous identification of an entity. One title designates only one entity, but one entity may have more than one title. The title plays a key role in order to provide the horizontal addressing entities.
- Requirements: are needs defined in the establishment of the logical link (workspace), which represent also the capabilities that entities must support to make part of a communication.
- Capabilities: are features supported by entities in order to meet the communication needs for a particular purpose.
- Horizontal Addressing: is an addressing scheme independent of the physical location of network entities, without the need for bandwidth reservation, network segmentation or specific physical connections.
- *Domain Title Service* (DTS): consists of a distributed system over the network [30], responsible for the maintenance of entity and provisioning of logical links required for communication. It is also able to understand their capabilities and needs, and for providing of features to treat them properly. Comprising *Domain Title Service Agents* (DTSA), it plays an important role in key aspects of the network, such as names and addresses, and have the ability to share the connection between the communicating entities. Throughout the network, DTSA are distributed in that domain being deployed at servers and network elements (switches, routers, and so on).
- Workspace: is a logical bus that has a title and contains network elements required to support the communication of the entities. The workspace is created by an entity that wants to communicate with a specific purpose. During its inception, the entity informs the set of requirements that must be supported by all entities who want to be part of the workspace. A new entity can be associated with an existing workspace and, if so, the logical bus can be extended to handle your communication. Likewise, an entity can move through the DTS being able to maintain it communicating. The main concept introduced by workspace is that the destination address is its title. Another important concept is that primitive, for example a stream, is sent once by the source and can be received by all the entities sharing it.
- DL-Ontology: is a logical link layer, able to semantically interpret and meet the requirements of the upper layers, using the infrastructure of the network optimally. It is the realization of logical link concept, being responsible for delivering data to the entities that compose the workspace.
- Net-Ontology: is responsible to semantically interpret the needs of the entities, and implement them through the DL-Ontology layer. It is a mechanism for semantic reasoning and features modularization, which links requirements and capabilities, establishing communication according to entities needs.
- *Entity Title Control Protocol* (ETCP): is a protocol that defines the communication between entities and DTS. It provides maintenance services of the entity and management of workspaces services. Example: entity-register, workspace-attach.
- *Domain Title Service Control Protocol* (DTSCP): is a protocol that defines the communication between DTSA's. Provides workspace search and register inter-DTSA services. Example: workspace-register and workspace-lookup.

One the main points of ETArch is the *horizontal addressing*, which solves the problem of ambiguity between identification and localization of the current architecture. In this approach, the identification of the *entity* is defined by its *titles*, and its localization is controlled by the *DTS* Agent immediately superior. When an *entity* wants to communicate, it creates a *workspace* by sending an ETCP message to DTSA. This *workspace* has a set of *requirements* which must match with the *capabilities* of the *entities* that wants to communicate. All the data transmitted over the network is delivered by the DL-Ontology, which is the main protocol of this architecture. It may be necessary to perform some additional processing by the network elements and hosts during the interpretation of the Net-Ontology, which defines the communication requirements. All communication is orchestrated by DTS, which is a distributed system materialized by their agents that communicate via DTSCP protocol.

IV. Implementation

This section aims to present an implementation scenario as a *proof of concepts*, regarding *workspace* concept applied to achieve the goal of *multicast aggregation*. We are mainly interested in observing the behavior of the network in the face of features like multicast, provided naturally by the architecture.

In order to overcome the existing limitations in the TCP/UDP/IP, including underlying protocols such as Ethernet and others, we developed a network interface which provides for the entities in a distributed environment free from legacy Internet protocols.

ETArch proposes a division between data and control planes, as well as OpenFlow, and its main components are: the DL-Ontology and Net-Ontology layers (in the data plane), and the DTS with its agents (in the control plane). The following sections describe how these components were implemented.

A. Net/DL-Ontology

The implementation has four main modules designed to have high cohesion and loose coupling for the entire workspace enrollment project.

The Ontology module is responsible for the design Title Model including the concepts of: DTS, Workspace, DL-Ontology, Title and Entities. It was modeled with software Protégé [31], and generated in *Ontology Web Language* (OWL) by using *OWL API* [32].

The module responsible for interpretation of OWL is under construction by the use of Jena. The reasoning of the ontology is a central point of the semantic approach, since it makes possible the creation of inference rules to implement the intelligence of Title Model. At this step, a parser based on regular expressions was used, and in the next stage of implementation will be included the reasoning.

The interface module is the implementation of a Java API for use by the parser and reasoning module. This includes communication through Raw Socket API, built in C language.

The Physical Medium Access module is responsible for the communication with physical layer allowing the primitive DL-Ontology to be sent to the physical environment without Internet protocols, such as Ethernet, IP, TCP, UDP, or SCTP.

B. DTSA (as an Openflow Controller)

As the DTSA's task of coordinating network elements is closely related to that of managing flows by an OpenFlow controller, we have decided to implement the first on top of the latter. In a nutshell, we extended the FloodLight open-source OpenFlow controller [33] to closely work with the DTSA.

The extensions to the Floodlight controller consisted in a new module that instantiates the DTSA and handles the exchange of DTS control messages.

As a extented *IOFMessageListener*, this module is able to listen incoming messages. By default, all messages that do not match any of the rules in the switch flow table are sent to the DTSA. When a message is received, the listener is called and checks if the message is a defined primitive. If so, the message is delivered to the DTSA which process it and modify the switches using a *flow_mod* [34].

V. Experiments

To experiment and evaluate the Entity Title Architecture, especially the workspace concept with its multicast capabilities, we conducted some experiments.

A simple topology, as shown in Figure 2, was defined. On the right side, a server contains a video application that produces a flow-based *Motion Joint Photographic Experts Group* (MJPEG). On the left side, at a host, one or more clients where instantiated during the experiments. Between the the hosts are three OpenFlow switches. Entities hosted at any host, including DTS Agent, are able to send and receive DL-Ontology primitives. Although, it is a simple topology that reflects a common situation where a server and a client are separated by a set of switches. The topology was created using Mininet [35], a system for rapid prototyping of OpenFlow-based networks.

To compare the Entity Title Architecture and the use of TCP/IP architecture for the networking, three different server applications where created. The first and the second ones based on UDP and IP protocols with *unicast* and *multicast* approaches respectively, and the third one based on our approach. Essentially, these applications are the same, and the main difference between them is just the way sockets are created and used.

At the application layer, a *Real-time Transport Protocol* (RTP) [36] based message is created, then, in the first case, Datagram Socket is used to send this message. The second

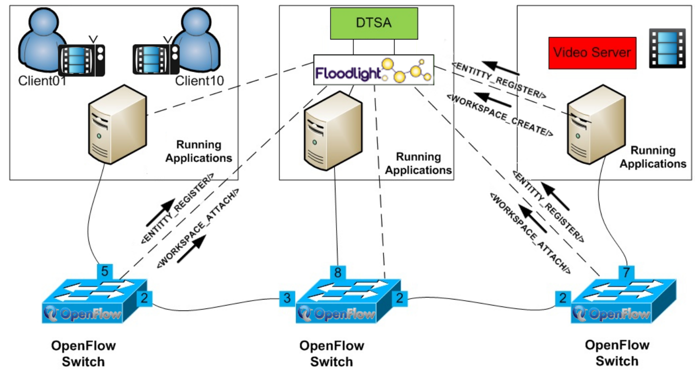

Figure 2. Scenario used for experimentation.

video application, that uses the workspace, creates a *Finsocket*, which is based on Raw Sockets. Raw Sockets does not use the TCP/IP stack and directly creates a frame and send it over the physical medium. In fact, the *Finsocket* does create a frame based on the Ethernet frame, but it does not contain the traditional information in its headers. Instead, the source address contains the leftmost bits of the workspace title and the destination address field is the rightmost bit.

Additionally, a management application for the DTSA was conceived, to allow a better visualization of the proposed scenario, as shown in Figure 3. Also, in this figure, one can observe two video subscribers attached to the workspace, and so, receiving the same flow.

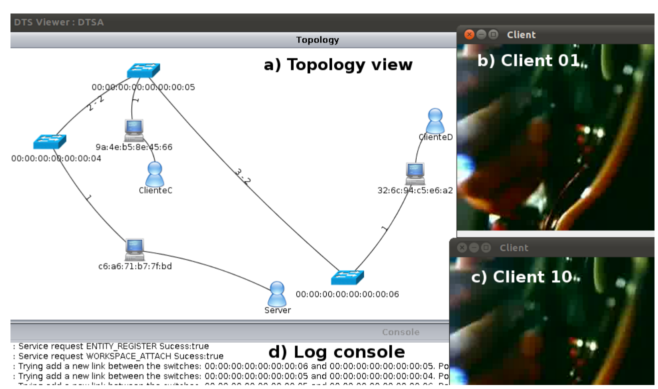

Figure 3. DTSA Management Application and Clients attached to the workspace.

VI. Results

At the experiments, a server application has been started and a different number of clients connected to it, requesting data. Considering the UDP/IP Unicast server application, in proportion as the number of clients grows, there is also an increase in bandwidth usage caused by the data replication on various flows instantiated. The video server that uses the Entity Title Architecture remains with a constant use of bandwidth at the source, no matter the number of clients. This is because the data is sent to workspace and a client connects to it, not directly to the server. The same occurs with the application

Copyright (c) IARIA, 2014.

using the UDP/IP Multicast, because the IP Multicast groups concept is related to ETArch workspace. In both approaches the data is replicated in the network elements; however, the IP Multicast has problems that make it unfeasible in global proportions for practical purposes [37]. Figure 5 shows the use of the bandwidth obtained from the comparison between the IP Unicast, IP Multicast and DL-Ontology approaches.

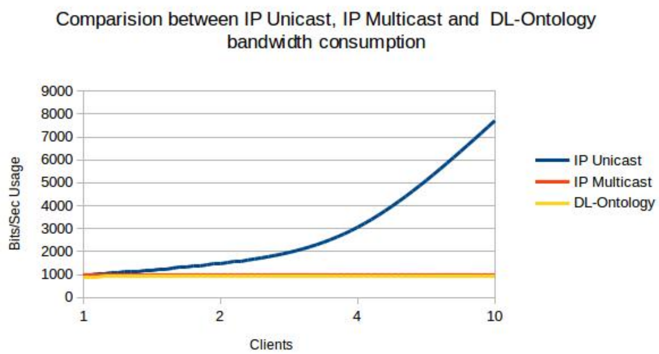

Figure 4. Bandwidth usage at the source versus the number of clients between all tested approaches.

Figure 5 focus on IP Multicast and DL-Ontology comparison. The results are similar; however, the DL-Ontology approach uses slightly less bandwidth, given the change of protocols used in the network and transport layer.

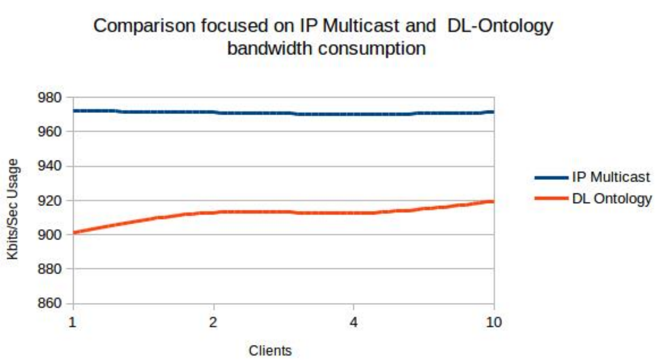

Figure 5. Bandwidth usage at the source versus the number of clients between IP Multicast and DL-Ontology approaches.

Although the results between the proposed and the conventional *multicast* are similar, the major advantage of our proposal is the possibility of deployment on a global scale by taking advantage of SDN. The ETArch approach proposes a new model for the Internet that gives natural support for *multicast* communication through drastic changes in aspects of addressing, identification and routing. The IP Multicast has limitations at: addressing, because of the limited number of *multicast* addresses, restricted to *class D*; network supporting, since it is necessary that all core devices provide this service; and control signaling, which imposes an impractical overhead in global scales.

VII. Conclusion

Considering the new set of requirements, the Internet architecture must be revised. This review process using a *clean slate* can free researchers from current deficiencies by providing a rich environment for experimentation.

In this article, we presented a SDN-based implementation of the *Entity Title Architecture*, and its application to address the multicast requirement. This work focused on the presentation of the main concepts of the architecture, demonstrating that the aggregation of multicast becomes a trivial task, because it is something intrinsic to the architecture.

Although OpenFlow can be used to implement the new naming, routing and addressing schemes, the literature on the topic does not contain detailed descriptions of how this can be done and this work aims to contribute in this matter too. So in addition to experimentally demonstrate the Tile Entity Architecture, this works also shows how an IP centered OpenFlow switch, compatible with OpenFlow 1.0 specification, can be used in networks that completely drop the TCP/IP stack from the data plane using a new semantics the for flow table.

The evaluation of the implemented architecture showed that the bandwidth used for the source remains constant regardless of the number of customers connected to it. The impact of this fact is that real connections can be used to provide services, such as high definition videos with efficient power consumption.

This was an expected result, because the Entity Title Architecture is based on a new naming and addressing scheme, where the destination address is the workspace and while the packet is sent to it, all entities that are part of it receives this packet bringing the architecture a seamless multicast capability. The workspace also provides mobility, cause it can move between the switchs, and in the presence of this event, the flow table will be automatically updated.

The approach presented in this paper is a more efficient form of communication if compared to the current solutions, such as IP Multicast (at network layer) and ALM (at application layer), by not having the limitations of TCP/IP architecture as demonstrated in this work. The ETArch provides a real *multicast* by drastic changes in routing and addressing schemes. There is no data repetition in the communication within the workspace, cause it provides a natural support to that requirement, differently from the ALM, which despite reduces the replication level, does not eliminate it completely, by presenting a strategy that does not take into account the access and distribution elements, just the core elements. Unlike IP Multicast, in ETArch approach it is possible for a host be attached to more than one workspace at the same time, through the flexibility in the routing rules provided by this architecture.

We are currently working on improving the security, routing and control plane aspects, which should be subject of the future work.

The results show that we are facing a viable approach to bring richer and more efficient services to the network, collaborating with research aimed to define, design and implement the next generation of computer network architectures.

ACKNOWLEDGMENT

This project was made under the support and belief of MEHAR team, valuable working partners. Therefore, we would like to thank all who contributed in any way to complete this stage of our research.

REFERENCES

[1] P. Baran, "On distributed communications networks," *IEEE Transactions on Communications Systems*, vol. 12, no. 1, pp. 1–9, Mar. 1964.

[2] V. Cerf and R. Kahn, "A protocol for packet network intercommunication," *Communications, IEEE Transactions on*, vol. 22, no. 5, pp. 637–648, 1974.

[3] J. H. De Souza Pereira, S. T. Kofuji, and P. F. Rosa, "Distributed systems ontology," in *New Technologies, Mobility and Security (NTMS), 2009 3rd International Conference on*, 2010, pp. 1–5.

[4] T. Zahariadis *et al.*, "Towards a future internet architecture," in *The Future Internet. Future Internet Assembly 2011: Achievements and Technological Promises*, ser. LNCS, J. Domingue, A. Galis, A. Gavras, T. Zahariadis, and D. Lambert, Eds. Berlin, Heidelberg: Springer-Verlag, 2011, vol. 6656, pp. 7–18. [Online]. Available: http://www.springerlink.com/content/978-3-642-20897-3#section=881237&page=15&locus=86 [retrieved: May. 2014]

[5] D. Farinacci, D. Lewis, D. Meyer, and V. Fuller, "The Locator/ID separation protocol (LISP)." [Online]. Available: http://tools.ietf.org/html/rfc6830 [retrieved: May. 2014]

[6] S. Deering, *Host extensions for IP multicasting*, ser. Request for Comments. IETF, Aug. 1989, no. 1112, published: RFC 1112 (Standard) Updated by RFC 2236. [Online]. Available: http://www.ietf.org/rfc/rfc1112.txt [retrieved: May. 2014]

[7] Y.-h. Chu, S. Rao, S. Seshan, and H. Zhang, "A case for end system multicast," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 8, pp. 1456 – 1471, Oct. 2002.

[8] J. Roberts, "The clean-slate approach to future internet design: a survey of research initiatives," *annals of telecommunications - annales des tlcommunications*, vol. 64, no. 5-6, pp. 271–276, May 2009. [Online]. Available: http://www.springerlink.com/content/e240776641607136/ [retrieved: May. 2014]

[9] F. de Oliveira Silva *et al.*, "Semantically enriched services to understand the need of entities," in *The Future Internet*, ser. Lecture Notes in Computer Science, F. lvarez *et al.*, Eds. Springer Berlin / Heidelberg, 2012, vol. 7281, pp. 142–153. [Online]. Available: http://www.springerlink.com/content/1222874ul734676k/abstract/ [retrieved: May. 2014]

[10] N. McKeown *et al.*, "OpenFlow: enabling innovation in campus networks," *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Mar. 2008, ACM ID: 1355746.

[11] G. Goth, "Software-Defined networking could shake up more than packets," *IEEE Internet Computing*, vol. 15, no. 4, pp. 6–9, Aug. 2011.

[12] K. Greene, "TR10: Software-Defined networking," *MIT Technology Review*, vol. 112, no. 2, Apr. 2009. [Online]. Available: http://www.technologyreview.com/web/22120/ [retrieved: May. 2014]

[13] C. Rothenberg *et al.*, "Revisiting IP Routing Control Platforms with OpenFlow-based Software-Defined Networks," *XXX SBRC - III Workshop de Pesquisa Experimental da Internet do Futuro-WPEIF*, p. 6, 2012.

[14] A. Dias *et al.*, "Cross Layers Semantic Experimentation for Future Internet," *XXX SBRC - III Workshop de Pesquisa Experimental da Internet do Futuro-WPEIF*, p. 16, 2012.

[15] N. Foster *et al.*, "Languages for Software-Defined Networks," *IEEE Communications Magazine*, p. 128, 2013.

[16] H. Kim and N. Feamster, "Improving Network Management with Software Defined Networking," *IEEE Communications Magazine*, p. 114, 2013.

[17] E. Commission, "The network of the future - projects," http://cordis.europa.eu/fp7/ict/future-networks/projects_en.html, 2012. [Online]. Available: http://cordis.europa.eu/fp7/ict/future-networks/projects_en.html [retrieved: May. 2014]

[18] M. DAmbrosio, M. Marchisio, V. Vercellone, B. Ahlgren, and C. Dannewitz, "4WARD. second NetInf architecture description," 2010. [Online]. Available: http://www.4ward-project.eu/index.php?s=file_download&id=70 [retrieved: May. 2014]

[19] J. H. d. S. Pereira, F. d. O. Silva, E. Lopes Filho, S. T. Kofuji, and P. F. Rosa, "Title model ontology for future internet networks," in *Future Internet Assembly 2011: Achievements and Technological Promises*. Springer-Verlag, 2011, vol. 6656, p. 465.

[20] N. S. Foundation, "NSF future internet architecture project," http://www.nets-fia.net/, 2011. [Online]. Available: http://www.nets-fia.net/ [retrieved: May. 2014]

[21] I. Seskar, K. Nagaraja, S. Nelson, and D. Raychaudhuri, "MobilityFirst future internet architecture project," in *Proceedings of the 7th Asian Internet Engineering Conference*, ser. AINTEC '11. New York, NY, USA: ACM, 2011, pp. 1–3. [Online]. Available: http://doi.acm.org/10.1145/2089016.2089017 [retrieved: May. 2014]

[22] C. Diot, B. Levine, B. Lyles, H. Kassem, and D. Balensiefen, "Deployment issues for the IP multicast service and architecture," *IEEE Network*, vol. 14, no. 1, pp. 78 –88, Feb. 2000.

[23] R. Hinden and S. Deering, *IPv6 Multicast Address Assignments*, ser. Request for Comments. IETF, Jul. 1998, no. 2375, published: RFC 2375 (Informational). [Online]. Available: http://www.ietf.org/rfc/rfc2375.txt [retrieved: May. 2014]

[24] D. Thaler, M. Handley, and D. Estrin, *The Internet Multicast Address Allocation Architecture*, ser. Request for Comments. IETF, Sep. 2000, no. 2908, published: RFC 2908 (Historic) Obsoleted by RFC 6308. [Online]. Available: http://www.ietf.org/rfc/rfc2908.txt [retrieved: May. 2014]

[25] R. Hinden and S. Deering, *IP Version 6 Addressing Architecture*, ser. Request for Comments. IETF, Feb. 2006, no. 4291, published: RFC 4291 (Draft Standard) Updated by RFCs 5952, 6052. [Online]. Available: http://www.ietf.org/rfc/rfc4291.txt [retrieved: May. 2014]

[26] E. Davies, S. Krishnan, and P. Savola, *IPv6 Transition/Co-existence Security Considerations*, ser. Request for Comments. IETF, Sep. 2007, no. 4942, published: RFC 4942 (Informational). [Online]. Available: http://www.ietf.org/rfc/rfc4942.txt [retrieved: May. 2014]

[27] I. Romdhani, M. Kellil, H.-Y. Lach, A. Bouabdallah, and H. Bettahar, "IP mobile multicast: Challenges and solutions," *IEEE Communications Surveys Tutorials*, vol. 6, no. 1, pp. 18–41, 2004.

[28] W.-P. K. Yiu and S.-H. G. Chan, "Offering data confidentiality for multimedia overlay multicast: Design and analysis," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 5, no. 2, pp. 13:1–13:23, Nov. 2008. [Online]. Available: http://doi.acm.org/10.1145/1413862.1413866 [retrieved: May. 2014]

[29] M. Hosseini, D. T. Ahmed, S. Shirmohammadi, and N. D. Georganas, "A survey of application-layer multicast protocols," *Commun. Surveys Tuts.*, vol. 9, no. 3, pp. 58–74, Jul. 2007. [Online]. Available: http://dx.doi.org/10.1109/COMST.2007.4317616 [retrieved: May. 2014]

[30] J. H. de Souza Pereira, S. T. Kofuji, and P. F. Rosa, "Horizontal addressing by title in a next generation internet," in *2010 Sixth International Conference on Networking and Services (ICNS)*. IEEE, Mar. 2010, pp. 7–11.

[31] M. Horridge, M. Musen, C. Nyulas, S. Tu, and T. Tudorache. (2012, May) protg. [Online]. Available: http://protege.stanford.edu/ [retrieved: May. 2014]

[32] M. Horridge, S. Bechhofer, and O. Noppens, "Igniting the OWL 1.1 touch paper: The OWL API." in *OWLED*, vol. 258. Citeseer, pp. 6–7. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.97.4920&rep=rep1&type=pdf [retrieved: May. 2014]

[33] Big Switch, "Floodlight OpenFlow controller," http://floodlight.openflowhub.org/, Jan. 2012. [Online]. Available: http://floodlight.openflowhub.org/ [retrieved: May. 2014]

[34] F. de Oliveira Silva, M. Goncalves, J. de Souza Pereira, R. Pasquini, P. Rosa, and S. Kofuji, "On the analysis of multicast traffic over the entity title architecture," in *2012 18th IEEE International Conference on Networks (ICON)*, p. 3035.

[35] B. Lantz, B. Heller, and N. McKeown, "A network in a laptop: rapid prototyping for software-defined networks," *Proceedings of the Ninth ACM SIGCOMM Workshop on Hot Topics in Networks*, pp. 19:1–19:6, 2010, ACM ID: 1868466.

[36] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, *RTP: A Transport Protocol for Real-Time Applications*, ser. Request for Comments. IETF, Jul. 2003, no. 3550, published: RFC 3550 (Standard) Updated by RFCs 5506, 5761, 6051. [Online]. Available: http://www.ietf.org/rfc/rfc3550.txt [retrieved: May. 2014]

[37] A. Boudani and B. Cousin, "SEM: a new small group multicast routing protocol," in *10th International Conference on Telecommunications, 2003. ICT 2003*, vol. 1, pp. 450–455.