

IEEE 802.21-enabled Entity Title Architecture for Handover Optimization

Carlos Guimarães*, Daniel Corujo*, Flávio Silva†, Pedro Frosi†, Augusto Neto‡, Rui L. Aguiar*

*Instituto de Telecomunicações, Universidade de Aveiro, Portugal

Email: {carlos.guimaraes,dcorujo,ruilaa}@ua.pt

†Faculdade de Computação, Universidade Federal de Uberlândia, Brazil

Email: {flavio,frosi}@facom.ufu.br

‡Universidade Federal do Rio Grande do Norte (UFRN), Brazil

Email: augusto@dimap.ufrn.br

Abstract—The explosion of demanding on-line services, such as video, is increasingly stressing the Internet architecture, requiring new solutions to support future usage scenarios. New Future Internet approaches targeting Information Centric Networking, such as the Entity Title Architecture (ETArch), provide new optimizations for these scenarios, using novel mechanisms such as Software Defined Networking (SDN). However, these are not targeted to address another growing Internet challenge reflecting the increase in multi-technology wireless mobile access. In this work, we empowered ETArch with Media-Independent mechanisms from the IEEE 802.21 standard, optimizing content delivery in wireless mobile scenarios. The resulting framework was implemented in a physical testbed of the OFELIA project, with results showing that the defined mechanisms supported seamless handover avoiding packet loss at a reduced overhead cost, when compared with the base ETArch. Moreover, the media-independent nature of these enhancements allows the framework to be deployed in different access technologies, in a flexible way.

I. INTRODUCTION

The Internet has been constantly evolving, motivated not only by its natural growth, but also by the introduction of new services and applications to fulfill emerging needs, leading to new requirements being placed over its architecture, such as mobility, security and scalable content distribution. To cope with this new set of requirements and to allow its evolution, several enhancements started to be defined, increasing the complexity of the overall Internet architecture, with many core components reaching their limit, and hindering further evolutions [1]. In addition, many current and emerging requirements still cannot be addressed adequately by the current Internet. As a result, new clean-slate architectures for the Future Internet are being proposed as the next step towards better solutions for current and future Internet utilization requirements.

Information Centric Networking (ICN) [2] is one of such proposed architectures, focusing on content access and delivery, improving over current host-to-host communications. Content has a more central role in the network operations, motivated by the need to meet data-intensive applications. This paradigm shift leverages in-networking caching and replication, improving efficiency, scalability and robustness. The Entity Title Architecture (ETArch) [3] is one of the proposed architectures that share the vision of content-oriented paradigms, where entities request content by subscribing it, triggering the network to dynamically configure itself in order

to provide the users with the content. In addition, it allows communicating entities to express their requirements over time. However deploying ICN capable nodes into current networks would require the update or replacement of existing networking equipment. Software Defined Networking (SDN) [4] emerges as a promising solution to overcome this, since it could not only facilitate the deployment of ICN functionalities in current networks without requiring new clean-slate designs, but it could also improve and enhance current and future Internet network management mechanisms.

Incrementally, the increasing proliferation of mobile devices equipped with multiple wireless interfaces (such as smartphones) creates complex heterogeneous scenarios where network operators need to provide connectivity for different kinds of access networks, to explore and provide an ubiquitous always best connected experience to the user. Furthermore, the traffic generated by mobile devices is increasing at an exponential rate and, by 2016, most of the traffic will be generated by applications related with mobile video consumed and/or produced by smartphones [5]. In this way, the capability of handling mobility efficiently in these future networks is a key requirement. The dynamics of the wireless environments, and the associated heterogeneity, pose a complex challenge for assuring that the content reaches the user in an optimized way.

As such, not only enhanced content-reaching procedures are needed (such as using a more content-oriented approach), but also supportive measures ensuring that the content is sent to the requester in the most optimal way (i.e., selected rate, selected codec, selected connection/technology/interface). Furthermore, aspects related to the characteristics of the content, as well as the current conditions of the wireless network, are all factors that can help to decide how to best deliver the content to the user. Considering this scenario, IEEE developed the IEEE 802.21 standard for Media Independent Handover (MIH) [6] [7]. Its main purpose is to facilitate and optimize inter-technology handover processes by providing a set of media-independent primitives for obtaining link information and controlling link behavior in a heterogeneous way, thus creating an abstraction regarding the link layer.

This work presents a core contribution from EDOBRA, Extending and Deploying OFELIA in Brazil, a workpackage of the OFELIA project¹, composing the integration of ETArch

¹FP7 OFELIA, <http://www.fp7-ofelia.eu/>

with IEEE 802.21. This integration empowers ETArch with a common control for wired and wireless networks, enabling not only the optimization of several network aspects (such as mobility and resources optimization), but also paves the way for a new set of scenarios, fueled by SDN mechanisms.

The remainder of the document is organized as follows: Section II presents the background for this work, highlighting not only the supporting technologies, but also other related approaches. Section III presents the proposed framework, evaluated in Section IV, where results of its deployment over OFELIA testbed are presented. Finally, the paper concludes in Section V and points out future work.

II. BACKGROUND

The Entity Title Architecture [3] is a clean slate network architecture, where naming and addressing schemes are based on a topology-independent designation that uniquely identifies an entity, called Title, and on the definition of a channel that gathers multiple communication entities, called Workspace. A key component of this architecture is the Domain Title Service (DTS), which deals with all control aspects of the network. The DTS is composed by Domain Title Service Agents (DTSAs), which maintain information about entities registered in the domain and the workspaces that they are subscribed to, aiming to configure the network devices to implement the workspaces and to allow data to reach every subscribed entity.

The operation of ETArch, on which a centralized entity is responsible for the behavior of the forwarding plane, meets SDN concepts, implemented in ETArch by the OpenFlow. OpenFlow [8] is an instantiation of SDN already available in a number of commercial products and used in several research projects. It separates the data plane from the control plane of the network, allowing a separate entity (i.e. the OpenFlow Controller) to manage and control the underlying data plane, configuring the forwarding table of the switches, via a well-known service-oriented API. This enables switches to be (re)configured on the fly, enabling flexible and dynamic network management [9].

The adoption of OpenFlow is mainly focused on core/wired networks. Thus, the support of mobility procedures in OpenFlow-enabled wireless environments (and therefore in ETArch) is not clearly defined, appearing only as solutions that consider SDN as an enabler mechanism for wireless protocols experimentation. Moreover, they have some limitations, such as disregarding network resources management, optimization or handover procedures support. [10] and [11] exposed OpenFlow to wireless environments, paving the way to support this innovation into wireless mobile networks. [12] deployed OpenFlow as an application on top of OpenWrt², providing a new set of experiment possibilities to be done over OpenFlow-enabled testbeds. In this way, in order to better support and manage network procedures in mobile wireless environments, media independent handover mechanisms, such as the ones from IEEE 802.21 are needed.

IEEE 802.21 [6] is an IEEE standard, whose main purpose is to define extensible media access independent mechanisms that facilitate and optimize handovers between heterogeneous

technologies, including IEEE 802 (both wired and wireless) and cellular technologies (e.g., 3GPP and 3GPP2). It encompasses aspects related to handover optimization by providing a set of services that allow the provisioning of link layer information (such as the status and condition changes of the link layer), to instruct configuration and handover related procedures at the link layers and to enhance handover decision processes with network configuration information about surrounding handover candidate networks. There is a number of existing attempts that integrated IEEE 802.21 with OpenFlow to support mobility procedures. The work described in [13] proposes an OpenFlow-based architecture for Wireless Mesh Networks that uses IEEE 802.21 to query link information and to trigger handovers. However, the details about the integration of both protocols are not clear, with the use of IEEE 802.21 limited to the discovery of association opportunities and to trigger the handover process in the Mobile Node (MN). This issue was progressed in [14], by proposing a framework that couples both mechanisms for dynamic optimized support of OpenFlow path establishment and wireless connectivity establishment, but not in an ICN environment.

In what concerns the ETArch itself, our previous work [15] presented a conceptual study about the integration of ETArch with IEEE 802.21, depicting the overall framework and its use in a mobility scenario. In the present work, we go even further by describing with more detail the components and the main workflows, integrating ETArch and IEEE 802.21, and evaluating it in the OFELIA experimental testbed in Brazil.

III. JOINT DTS AND IEEE 802.21-ENABLED OPENFLOW FRAMEWORK

Our framework aims to enhance ETArch architecture with mechanisms and capabilities provided by IEEE 802.21, facilitating and optimizing horizontal and vertical handovers, by enhancing the information exchanged between network entities and MNs. In addition, the network controlling entities can also be aware of the current MN link connection and other available wireless networks detected by MNs, including their link parameters and conditions, and use this information to optimize handover processes. Likewise, by using OpenFlow capabilities, controlling entities can use this information to (re)configure the flows across the whole network, optimizing network resources. As such, using our framework, ETArch can target a whole new set of network management scenarios, enabled and supported by the information provided by IEEE 802.21. For example, DTS workspaces can be dynamically and preemptively configured according to handover candidates detected by the MN while it moves, mitigating the effects of the handover procedure. Finally, our framework envisages the configuration of OpenFlow flows up to the network side endpoint, such as wireless network point of attachments (e.g. WLAN Access Point). Although, in this work, these mechanisms focus mostly on WLAN technologies, they could be applied to other wireless (such as LTE and WiMAX) and wired (such as Ethernet) technologies as well, due to the technology agnostic capability provided by IEEE 802.21.

The proposed framework is presented in Figure 1, depicting the enhancements made over the DTSA, the OpenFlow switch and the MN. The three main elements are described as follows:

²OpenWrt - <https://openwrt.org/>

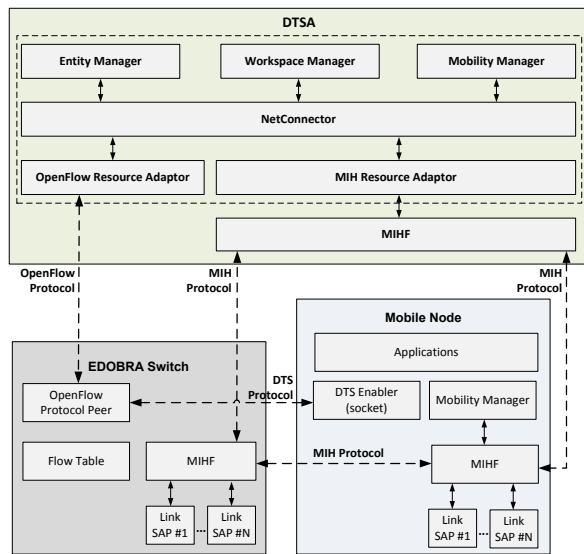


Fig. 1: Proposed framework

DTSA: The DTSA acts simultaneously as the OpenFlow controller and the Point of Service (PoS) of the network. In what concerns its functions as OpenFlow controller, the DTSA is responsible for both storing information about the existing entities (Entity Manager) and workspaces (Workspace Manager), as well as for performing routing related tasks, implementing the workspaces into the switches. Its function as PoS consist in handling and controlling mobility procedures (Mobility Manager). Moreover, these functions are interfaced by a central module (NetConnector), allowing the integration of procedures to optimize several aspects of the network. Lastly, it features a Media Independent Handover Function (MIHF) for exchanging IEEE 802.21 information with other nodes and an OpenFlow Channel for communication with the OpenFlow Switches.

EDOBRA Switch: The EDOBRA Switch consists of an IEEE 802.21-enabled OpenFlow switch. Besides the standard OpenFlow switch capabilities for executing data packet forwarding operations and for storing information on how packets of each workspace should be treated, the EDOBRA Switch is coupled with IEEE 802.21 mechanisms to control aspects of the link interface regarding handover management, such as resource management and/or events about the attachment and detachment of nodes. Lastly, it is coupled with an MIHF for interacting with the MN and the DTSA via IEEE 802.21 and an OpenFlow Channel for communication via OpenFlow with the DTSA. The OpenFlow Channel is also responsible for encapsulating DTS messages into OpenFlow messages.

Mobile Node: The MN represents the end-user equipment that establishes connection with the endpoint switches. The MN may be equipped with one or more access technologies, either wired (e.g., Ethernet) or wireless (e.g., WLAN or 3G). The MN deploys an MIHF, allowing higher-layer entities in the device itself (Mobility Manager) or external network entities (e.g., DTSA) to control the links and to retrieve information in an abstract way. In this way, the MN is able to either retrieve link conditions on the current connection or to provide information about other networks in its range. In what concerns

DTS procedures (such as register, workspace creation and attachment operations), the MN contains a DTS Enabler that allows it to communicate with endpoint switches via DTS. In addition, the DTS Enabler is also used by applications to send their packets over DTS protocol.

A. Handover Procedures Workflows

In the following, we present the generic workflows on the DTSA behavior that occur in different phases of the handover procedure, namely Handover Preparation, Handover Commit and Handover Complete phases.

The first phase of the handover process occurs before the handover action itself, in which different networks in the range of the MN are queried about their resources with the purpose of verifying if they can support the attachment of the MN (as described in Algorithm 1). This verification may include not only queries about the MN subscribed workspaces, but may also include queries on quality conditions and resources. Based on all this information, candidate networks are ordered (or removed from the list) and sent to the MN, which will decide if it will move to any of those networks based on the provided ordered list and on the link condition of each one.

Algorithm 1: Handover Preparation

- 1 Get all workspaces subscribed by the MN
 - 2 **for** each network in the range of the MN **do**
 - 3 Check available resources
 - 4 Order networks by preference order
 - 5 Return results to MN
-

Our framework supports preemptive workspace extension, aiming to mitigate the gap between the handover and the content reception in the MN (e.g. seamless handover). Thus, before the DTSA instructs the MN to handover to a new network, it must extend the MN subscribed workspaces towards the candidate PoA, as depicted in Algorithm 2. Based on the MAC address of the candidate PoA, the DTSA needs to learn the switch ID and corresponding port with the purpose of discovering the new location of the MN. Then, for each workspace subscribed by the MN, the DTSA associates the tuple (*switch ID, Port*) from the candidate PoA with the MN, allowing the DTSA to recalculate the workspace path up to the candidate PoA. If the referring PoA and output port are not yet in the workspace, the DTSA configures the workspace flow in every new switch in the path. After configuring all switches, the DTSA instructs the MN to handover to the candidate PoA.

Algorithm 2: Handover Commit

- 1 Discover switch ID and Port based on PoA MAC
 - 2 Get all workspaces subscribed by the MN
 - 3 **for** each workspace subscribed by the MN **do**
 - 4 Associate (*switch ID, Port*) to the MN
 - 5 Recalculate workspace path up to PoA
 - 6 **if** PoA and Port not in workspace **then**
 - 7 **for** each new switch in the path **do**
 - 8 Configure workspace flow using OpenFlow
 - 9 Instruct MN to handover to PoA
-

The last phase of the handover procedure is the completion, which may include the release of workspaces in the switches

that became unnecessary after the handover of the MN (Algorithm 3). For each workspace subscribed by the MN, the DTSA disassociates the MN from the tuple (*switch ID,Port*) of the old PoA, followed by the recalculation of the workspace. If the old PoA is removed from the workspace path, the DTSA removes the workspace flow from all switches that do not require it anymore, thus freeing resources.

Algorithm 3: Handover Complete

```

1 Get all workspaces subscribed by the MN
2 for each workspace subscribed by the MN do
3   Disassociate (switch ID,Port) from the MN
4   Recalculate workspace path
5   if path PoA removed from workspace then
6     for each superfluous switch in the path do
7       Remove workspace flow using OpenFlow
8 Report handover completion to MN

```

B. Use Case: MN-initiated handover scenario

Figure 2 depicts a scenario where a MN detects a network with better signal quality, triggering the required handover procedures to move to that network. The handover trigger could have different origins, such as detecting a non-optimal access technology preference of the MN towards video or even an higher packet error rate at the current link.

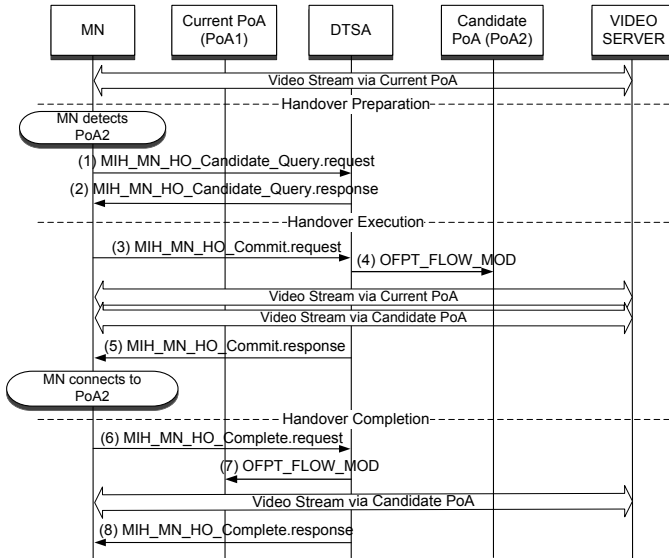


Fig. 2: MN-initiated handover scenario signaling

This scenario starts with the MN receiving a video feed through its serving network (PoA1), with the flow being generated in the video server. The MN, detecting a network with a better signal strength (i.e. the network belonging to the candidate PoA), issues a MN-initiated handover, using the MIH protocol, by sending a *MIH_MN_HO_Candidate_Query.request* message to its home DTSA (1). The DTSA verifies available resources in the candidate PoA and if it is able to accommodate the MN's subscribed workspaces. The resources required along the whole path are also verified. Then, it replies to the MN by sending a *MIH_MN_HO_Candidate_Query.response* message (2).

The MN verifies if the candidate PoA is still feasible and, if so, notifies the DTSA about the selected target network, sending a *MIH_MN_HO_Commit.request* message (3). The DTSA prepares the required resources in the candidate PoA to accommodate the MN and, if the workspaces subscribed by the MN are not configured up to the candidate PoA, it extends the workspaces up to PoA2 by sending *OFPT_FLOW_MOD* messages (4). This includes the configuration of the workspace in all on-path switches between the candidate PoA and the source. Then, the DTSA acknowledges the MN that the resources were reserved and that it can move to the new network (5).

Upon the reception of this message, the MN executes the attachment to the candidate PoA network and, since the workspace was preemptively extended to the new location of the MN, the MN immediately starts receiving the video stream. In parallel, after moving to the candidate PoA, the MN informs the DTSA of the handover result, by sending a *MIH_MN_HO_Complete.request* message (6). At this moment, the DTSA knows that the MN is no longer in the network of PoA1 and, if there are no other subscribers to same workspaces in the PoA1, it issues *OFPT_FLOW_MOD* messages to request the PoA1 to release the subscriptions to the workspaces associated with the MN (7). Finally, the DTSA acknowledges the MN about the conclusion of the handover procedures (8).

Although we present a MN-initiated handover scenario, our framework is flexible to cope with different handover strategies, such as network-initiated handover.

IV. EVALUATION

In order to evaluate the feasibility of our framework, we integrated ODTONE³ [7], an open-source IEEE 802.21 implementation, with the implementation of ETArch according to the proposals in Section III.

A. Testbed Description

Our evaluation scenario was built over the Brazilian island of the OFELIA testbed. As presented in Figure 3, two different PoAs (TP-Link TL-WR1043ND) were selected, which are connected to a common OpenFlow Switch (Datacom DM4000 ETH24GX+2x10GX). The PoAs are OpenFlow and IEEE 802.21-enabled. The DTSA is connected to the OpenFlow devices using two different connections: one for control and another for data. The MN1, MN2 and the Video Server, on which the DTS applications were run, are the remaining entities that complete the evaluation scenario. The application in the video server is sending a H.264 video stream over the "edobra" workspace, with the MN1 and the MN2 subscribed to that workspace in order to receive the video stream. The IEEE 802.21 messages are sent over DTS protocol, using the "odtone" workspace.

In this scenario, MN2 moves from PoA1 to the PoA2, while receiving content from the Video Server, using the signaling depicted in Figure 2 to update the workspace towards the new PoA, optimizing the handover process. Finally, it assumes that the MNs are initially attached to PoA1 and already receiving the content, initiating the handover to PoA2 at time $t=10s$ of the experiment. Each experiment was run 10 times, showing

³Open Dot Twenty ONE - <http://atnog.av.it.pt/odtone>

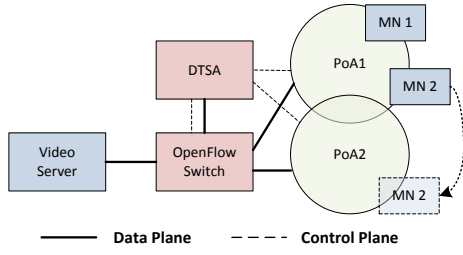


Fig. 3: Scenario description

here averaged results with a 95% T-Student confidence interval. In order to correlate the results, the handovers start at the same time between experiments.

B. Performance Evaluation

In this section, we evaluate the performance of the proposed framework, comparing it with a deployment of the ETArch without IEEE 802.21. Obtained results are shown in Table I, which presents the impact on content reception, including duration and number of duplicated packets (in the network and in the MN), lost packets during the handover and the time required to restore the stream after the handover. Figure 4 presents the content reception on the MN using each solution, focusing on the time that the handover occurs.

	ETArch with IEEE 802.21	ETArch only
Lost packets during HO	≈ 0	19.4 ± 1.0
Restore stream delay (ms)	≈ 0	407.9 ± 26.9
Redundancy at MN (ms)	40.3 ± 12.4	0
Redundancy at network (ms)	851.7 ± 32.1	0
Redundancy at network (packets)	39.6 ± 3.1	0

TABLE I: Content reception performance comparison

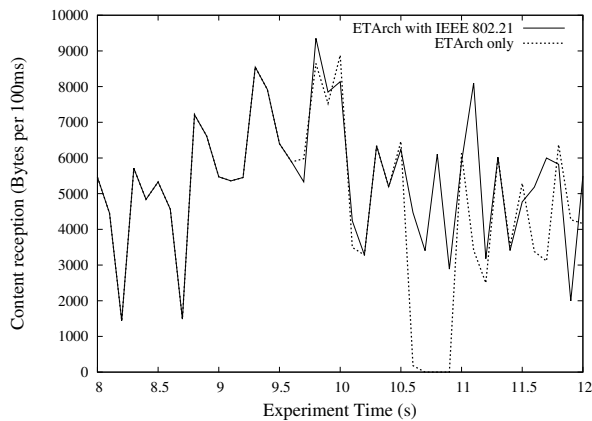


Fig. 4: Content reception on the MN

Analyzing the results of Table I, we can observe that, in the proposed scenario (i.e., make-before-break handover), our framework enabled the handover to occur with no packet loss. This was achieved because, during the handover procedure,

the MN only deactivated the connection to PoA1 when the connection with PoA2 was already established. Thus, during the handover procedure, the MN received the content from both interfaces during approximately 40 ms, not interrupting the content reception on the MN (as observed in Figure 4). Since the workspaces were previously extended, it resulted in no delay to restore the video stream. However, it means that packets from the video stream workspace were sent towards the new location, even if no subscribers were present, for about 851.7 ± 32.1 ms (which includes about 779.9 ± 29.3 ms of L2 handover time), representing approximately 40 packets.

Without using our framework, after the L2 handover process, the DTS application on the MN needed to unregister from the DTSA through the old PoA and to re-register through the new PoA. This register procedure took 407.9 ± 26.9 ms, during which time, the MN did not receive any packet (as observed in Figure 4), representing approximately 19.4 packets lost during the handover. In consequence of this behavior, there was no redundancy during the handover, in opposite to what occurs on our framework. Comparing our framework with this solution, we verified that our framework was able to reduce significantly the packet loss, by extending the workspaces to the new PoA before the handover, allowing it to receive the contents from both interfaces while moving across different networks. Moreover, with our framework, the handover process becomes transparent to the DTS application, since no DTS register and attach operations are needed to restore the workspace flow.

C. Control Signaling Overhead Analysis

In this section we study the footprint of each protocol in the proposed signaling. The results are presented in Table II, showing the amount of data exchanged for each protocol and the total time required for the different phases of the handover process. The values for the amount of data exchanged consider the size of the whole message.

	HO Preparation	HO Commit	HO Complete
Size (bytes)			
IEEE 802.21	235	171	218
OpenFlow	0	2568	0
DTS	0	0	0
Time (ms)	108.3 ± 18.5	65.7 ± 17.5	32.0 ± 10.9

TABLE II: Total signaling overhead per handover

Results from Table II show that almost 20% of the exchanged signaling corresponds to IEEE 802.21 protocol, with the remaining 80% related with the OpenFlow protocol. However, the IEEE 802.21 signaling does not depend on the number of workspaces subscribed by the MN, unlike the OpenFlow signaling. Therefore, the percentage of IEEE 802.21 overhead could be even lower if the MN was subscribed to more workspaces. In what concerns the DTS protocol, no control signaling was required during the handover procedure since DTS operations on DTSA were triggered by the IEEE 802.21 protocol. The signaling involving the MN accounted for about 20% of the total signaling overhead, due to its assistance in all phases of the handover process.

Comparing each phase, the handover commit was the most demanding phase, mainly due to the configuration of the workspace flows in the OpenFlow devices, corresponding to approximately 86% of the total signaling. In our scenario, two workspaces were reconfigured: the video stream workspace and the IEEE 802.21 control workspace. Thus, each workspace that the MN was subscribed increased the amount of OpenFlow signaling in 1284 bytes. In what concerns the complete phase, another MN was attached to the PoA1 and, therefore, the workspace was not removed from PoA1 during the handover complete phase. However, if the MN was the only subscriber to the workspaces on PoA1, the DTSA would have triggered the mechanisms to remove the workspaces from PoA1, increasing the overhead introduced by the OpenFlow protocol (about 1268 bytes per workspace).

In terms of delay, the handover preparation procedures are the most demanding phase (about 108.3 ± 18.5 ms). It encompasses the time to compute the available resources on each available PoA, with the purpose of finding the best handover candidate for the MN. In our scenario, the commit procedures required more time than the complete procedures because, since the MN1 remained in the PoA1, there was no reconfiguration of the workspaces on PoA1 after the handover. Still, the overhead in terms of time introduced by the IEEE 802.21 procedures accounts only to 21% of the total handover time. The remaining 79% respects the L2 handover procedure, which took 779.9 ± 29.3 ms. The L2 handover procedure includes the scanning, association and authentication steps.

V. CONCLUSION AND FUTURE WORK

We have presented an IEEE 802.21-enabled framework that aims to support optimized media independent mobility procedures over ETArch, a clean-slate SDN-based ICN approach. It allows the dynamic and preemptive reconfiguration of the network using information about possible handover candidates and link conditions perceived by the MN, mitigating the impact to on-going sessions on the MN. This framework was deployed over the OFELIA testbed, featuring a mobility scenario. Results showed that our framework allows avoiding packet loss, while achieving an acceptable overhead resulting from the new mobility optimization messages, when compared with normal ETArch operations. In order to avoid this packet loss and achieve seamless handover management, some redundancy of received packets in the new interface is encountered. However, our framework is flexible enough to encompass different handover management strategies (either controlled and/or assisted by the MN or the network), in order to surpass this issue. Thus, this flexibility further allows new scenarios to be considered, which are able to cater to different kinds of access technologies (in a media independent way), as well as different applications and utilization patterns. Moreover, using our framework, applications become unaware of the handover process. The work presented in this article showcased the integration and growth capabilities of multiple technologies, exposing them to novel scenarios, contribution to the evolution of SDN, ICN and mobility management procedures operating as a suitable Future Internet framework embodiment.

As future work, we are currently improving ETArch with Quality of Service control capabilities to enable optimized bandwidth-guaranteed transport, mobility and resilience.

ACKNOWLEDGMENT

This work has been partially funded by the European Community's Seventh Framework Programme, under grant agreement n. 258365 (OFELIA project), by the project Cloud Thinking (CENTRO-07-ST24-FEDER-002031), co-funded by QREN, Mais Centro program, and by Brazilian agencies CAPES and FAPEMIG.

REFERENCES

- [1] M. Handley, "Why the internet only just works," *BT Technology Journal*, vol. 24, no. 3, pp. 119–129, Jul. 2006. [Online]. Available: <http://dx.doi.org/10.1007/s10550-006-0084-z>
- [2] Information-Centric Networking Research Group (ICNRG). The internet engineering task force. [Online]. Available: <http://irtf.org/icnrg>
- [3] 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)*, 2012, pp. 30–35.
- [4] M.-K. Shin, K.-H. Nam, and H.-J. Kim, "Software-defined networking (sdn): A reference architecture and open apis," in *ICT Convergence (ICTC), 2012 International Conference on*, 2012, pp. 360–361.
- [5] CISCO. Cisco visual networking index: Global mobile data traffic forecast update, 20112016 [visual networking index (vni)] - cisco systems. [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.html
- [6] LAN/MAN Committee of the IEEE Computer Society, "IEEE Std 802.21-2008, Standards for Local and Metropolitan Area - Part 21: Media Independent Handover Services," 2008.
- [7] D. Corujo, C. Guimaraes, B. Santos, and R. L. Aguiar, "Using an open-source ieee 802.21 implementation for network-based localized mobility management," *Communications Magazine, IEEE*, vol. 49, no. 9, pp. 114–123, 2011.
- [8] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "Openflow: enabling innovation in campus networks," *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Mar. 2008. [Online]. Available: <http://doi.acm.org/10.1145/1355734.1355746>
- [9] H. Kim and N. Feamster, "Improving network management with software defined networking," *Communications Magazine, IEEE*, vol. 51, no. 2, pp. 114–119, 2013.
- [10] K.-K. Yap, R. Sherwood, M. Kobayashi, T.-Y. Huang, M. Chan, N. Handigol, N. McKeown, and G. Parulkar, "Blueprint for introducing innovation into wireless mobile networks," in *Proceedings of the second ACM SIGCOMM workshop on Virtualized infrastructure systems and architectures*, ser. VISA '10. New York, NY, USA: ACM, 2010, pp. 25–32. [Online]. Available: <http://doi.acm.org/10.1145/1851399.1851404>
- [11] K.-K. Yap, M. Kobayashi, R. Sherwood, T.-Y. Huang, M. Chan, N. Handigol, and N. McKeown, "Openroads: empowering research in mobile networks," *SIGCOMM Comput. Commun. Rev.*, vol. 40, no. 1, pp. 125–126, Jan. 2010. [Online]. Available: <http://doi.acm.org/10.1145/1672308.1672331>
- [12] Y. Yiakoumis, J. Schulz-Zander, and J. Zhu. (2012, Oct.) Pantou : Openflow 1.0 for openwrt @ONLINE. [Online]. Available: http://www.openflow.org/wk/index.php/Pantou:_OpenFlow_1.0_for_OpenWRT
- [13] P. Dely, A. Kessler, and N. Bayer, "Openflow for wireless mesh networks," in *Computer Communications and Networks (ICCCN), 2011 Proceedings of 20th International Conference on*, 2011, pp. 1–6.
- [14] C. Guimarães, D. Corujo, R. L. Aguiar, F. d. O. Silva, and P. Rosa, "Empowering software defined wireless networks through media independent handover management," in *Globecom 2013 - Next Generation Networking Symposium (GC13 NGN)*, Atlanta, USA, Dec. 2013.
- [15] F. d. O. Silva, D. Corujo, C. Guimarães, J. H. d. S. Pereira, P. F. R. Rosa, S. T. Kofuji, and R. Aguiar, "Enabling network mobility by using IEEE 802.21 integrated with the entity title architecture," in *XXI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos. Anais do IV Workshop de Pesquisa Experimental na Internet do Futuro (WPEIF)*. Brasília: Sociedade Brasileira de Computação, 2013, pp. 29–34.