A L I E N

ABSTRACTION LAYER FOR IMPLEMENTATION OF EXTENSIONS IN PROGRAMMABLE NETWORKS

Collaborative project co-funded by the European Commission within the Seventh Framework Programme

Grant agreement no: 317880
Project acronym: ALIEN
Project full title: "Abstraction Layer for Implementation of Extensions in programmable Networks"
Project start date: 01/10/12
Project duration: 24 months

Deliverable D4.1
Requirements for integration with the OFELIA facility

Version 1.0

Due date: 30/04/13

Submission date: 13/05/13

Deliverable leader: UPV/EHU

Author list: Jon Matias (UPV/EHU), Matteo Gerola (Create-Net), Damian Parniewicz (PSNC), Mehdi Rashidi (UoB), Eduardo Jacob (UPV/EHU), Maider Huarte (UPV/EHU), Elio Salvadori (Create-Net), Roberto Doriguzzi (Create-Net), Lukasz Ogrodowczyk (PSNC), Artur Juszczyk (PSNC) and Krzysztof Dombek (PSNC).

Dissemination Level		
☒	**PU:**	Public
☐	**PP:**	Restricted to other programme participants (including the Commission Services)
☐	**RE:**	Restricted to a group specified by the consortium (including the Commission Services)
☐	**CO:**	Confidential, only for members of the consortium (including the Commission Services)

Abstract

This deliverable analyses the integration of the ALIEN hardware carried out in this project under the OFELIA Control Framework (OCF) developed in the FP7 project OFELIA. First, the OFELIA Control Framework (OCF) is presented, both the current and future architecture, and the requirements imposed by this control framework are described in Section 2. Then, the ALIEN hardware available at this project is analysed and categorized in Section 3 to extract the requirements imposed by each platform. In this case, the plans each partner has for their platform could be relevant. After studying the deep implications of using OpenFlow version 1.0 in OFELIA, the impact on supporting different versions of OpenFlow in FlowVisor which is underlying in OCF and both sources of requirements, a Time-Based sharing is presented as a general approach to the ALIEN hardware integration under the OCF. The architecture of the solution and the specific modules to develop are identified in Section 4, as long as the functional description of those modules. The Time-Based sharing approach is OpenFlow agnostic, which means that the OpenFlow version is not inspected. Thus, any current or future version of the OpenFlow protocol or any private extension of them will be supported in the same way by means of this solution. Finally, some conclusions are described in Section 5.

Table of Contents

Figure Summary

Table Summary

Executive Summary

The current OFELIA experimental facility is oriented to enable the experimentation with OpenFlow at European level. Therefore, the OpenFlow infrastructure is exposed to the researchers in order to test novel networking approaches. Basically, OFELIA is a shared facility with two types of resources: (1) Virtual Machines (VMs) based on Xen; and (2) OpenFlow enabled switches. The researchers apply for both types of resources and obtain computational capacity at VMs and a control interface to the networking elements. In a nutshell, OFELIA allows researchers to not only experiment on a test network but to control the network itself.

OpenFlow (OF) has evolved quickly in the last years with several different versions (v1.0, optical extensions v0.3, v1.1, v1.2, v1.3, and the last version v1.3.1) in a short period of time, which in fact are incompatible between them. This situation causes some chaos and has been very demanding for vendors, developers and researchers. Currently, the most widely deployed and implemented version of OpenFlow (both by vendors and open source community) is OFv1.0. This is the reason why OFELIA has deployed OpenFlow version 1.0 (OFELIA started in October of 2011).

As an experimental facility, OFELIA needs to share its resources between several experiments and researchers. The OFELIA Control Framework (OCF) is in charge of the management of the whole testbed and its resources. The OCF also manages the life-cycle of the experiments and which resources are assigned to them. As previously mentioned, there are two types of resources to be shared: VMs and OpenFlow switches. On the one hand, the sharing of computational resources is a well-known technique with several commercial and open source products in widespread use. The virtualization software used by OFELIA is Xen. The OFELIA project has developed a mechanism to integrate the management of the VMs under the OCF. On the other hand, the sharing of OpenFlow switches is based on FlowVisor, which is a special OpenFlow Controller that acts as a transparent proxy between the resource (i.e. the switch) and multiple Controllers. The FlowVisor is an external entity, which is able to delegate parts of the flowspace of the switch to different Controllers and isolate the control plane associated with each part. Therefore, the FlowVisor allows the slicing and virtualization of the switches, enabling the sharing of OpenFlow switches between several experiments at the same time. OFELIA has defined the VLAN tag as the mechanism to enforce the isolation, that is, the field to isolate the flowspace between experiments. FlowVisor inspects the OpenFlow protocol to enforce the isolation between experiments, and consequently, it depends on the OpenFlow version. Currently, FlowVisor only supports OpenFlow version 1.0. This means that OFELIA only allows sharing OFv1.0 resources due to its tight relation with FlowVisor.

Nowadays, there are more and more OpenFlow resources (i.e. switches and controllers) that implement OpenFlow versions beyond 1.0. In the research community it is important to experiment with the latest tools and updated elements, therefore, OFELIA faces one limitation: how to test OpenFlow resources beyond v1.0 and integrate its control under the OCF. In fact, ALIEN hardware is a clear example of this restriction. The ALIEN hardware can implement

OpenFlow v1.3 or some particular extensions to v1.0 (e.g. an experiment oriented to develop and test those extensions). Since the FlowVisor needs to inspect the OpenFlow protocol and this is not the standard v1.0, the ALIEN hardware cannot be directly integrated under the OCF. In this case, the FlowVisor should be updated each time to the new version. Even, this version could be specific to the experiment. Due to the huge efforts needed and how particular the solution is (devoted to the specific version), this deliverable explores an alternative to the current solution at OCF.

This deliverable first analyses the requirements imposed both by the OFELIA Control Framework (Section 2) and the ALIEN hardware (Section 3). After this analysis a proposal for integrating the ALIEN hardware under the OCF is described, as well as its architecture based on those requirements (Section 4). The Time-Based sharing is an alternative to FlowVisor, which avoids inspection of the OpenFlow protocol. This characteristic makes the approach more general and future proof. Opposite to the FlowVisor solution, the Time-Based approach redirects the whole control (i.e. the OpenFlow control interface) to the adequate Controller and adapts the data plane to be consistent with the legacy infrastructure in OFELIA.

An alternative to the Time-Based sharing will be considered in the future. In this context, the internal slicing/virtualization of the OpenFlow Enabled Switches (opposite to the external slicing/virtualization done by the FlowVisor) will be analysed. Having a common implementation on top of the Hardware Abstraction Layer can help to implement this option.

1 Introduction

The main objective of the Work Package 4, and this Deliverable D4.1 in particular, is to deal with the integration of the ALIEN hardware developed in this project under the OFELIA Control Framework (OCF). This deliverable focuses on the requirements and specifications for achieving the integration of this ALIEN hardware, as well as the definition of a preliminary architecture and possible solution for its incorporation under the OCF.

In order to proceed with this integration, there are two different parts to analyse. On the one hand, the OFELIA Control Framework manages the life-cycle of experiments. It manages the resources assigned to each experiment in a particular moment. Currently, there are two types of resources shared by the OFELIA experimental facility: Virtual Machines (VMs) and OpenFlow Enabled Switches (with standard v1.0). Those resources are shared by means of different tools, the VMs make use of Xen [Xen] to share computational resources, whereas the FlowVisor is used to share the flowspace exposed by the OpenFlow switches available in the infrastructure between different experiments. On the other hand, the ALIEN hardware is a new set of OpenFlow oriented resources (with support for different OpenFlow versions or even extensions), that this project wants to share between the OFELIA experiments and researchers by using the same control framework currently deployed, the OCF.

When talking about the integration of ALIEN behind the OCF, this means that this hardware should be controlled and managed by the same OFELIA Control Framework used at OFELIA. This also means that the ALIEN hardware can be shared by the experiments running on OFELIA and, that its resources are exposed by the OCF tool to the researchers. In this context, it is important to clearly identify and extract the requirements that any of those parts can impose to this integration.

First of all, regarding the requirements imposed by the OFELIA Control Framework, some background should be introduced to contextualize them. Basically, the OCF manages the life-cycle of the experiments and is in charge of enabling the sharing of resources between those experiments. It is based on Expedient, which was originally designed to facilitate the integration of new resources of very different nature. Currently, there is one version of the OCF deployed at OFELIA facility, but a new version with some improvements is planned to be integrated before the end of the OFELIA project. All this aspects are analysed in Section 2, which ends with the list of the requirements imposed by the OCF to the integration of any type of resource under its control framework.

Secondly, concerning the requirements imposed by the ALIEN hardware to proceed with its integration under the OCF, it is important to understand the nature and type of hardware that this project is dealing with. Based on previous analysis done in the Deliverable D3.1 [ALIEN_D3.1], the hardware is categorized and analysed, and different alternatives and options for the future development in WP3 are considered. It is worth to mention, that each partner could have specific plans for their development, since some optional decisions could be taken. Some ALIEN hardware imposes very

restrictive requirements due to its own foundation. All these aspects are analysed in Section 3, which also proposes a list of requirements that the ALIEN hardware imposes for its integration under the OCF.

In the end, it is important to solve different aspects at different planes. This means that the proposed solutions must deal with the management, control and data planes. All these aspects should be properly integrated. The proposed solution should be compatible with the legacy OFELIA resources, which implies that it should be possible to integrate ALIEN resources with legacy resources in a common experiment. Currently, the OFELIA legacy resources are the VMs and the OpenFlow switches, and data plane compatibility is a big issue to solve.

Once the requirements are described, several alternatives and solutions have been suggested and discussed. As a result, a first approach to a possible architecture of the solution to integrate ALIEN hardware within the OCF is proposed in Section 4. The architecture is defined and details about its implementation are described. This definition is a preliminary work and the on-going work in WP4 could evolve this architecture.

In general, the most challenging part in this integration is the fact that the ALIEN hardware can implement different versions of OpenFlow (v1.0, optical extensions v0.3, v1.1, v1.2, v1.3 or v1.3.1) or even private extensions to those versions. However, this WP4 tries to find a general solution for all current selected ALIEN hardware (or even future developments). In order to find a general solution, the integration should be done without inspecting the OpenFlow protocol in the control framework. Otherwise, there should be one solution per OpenFlow version, which is in fact what has been done so far with the FlowVisor. FlowVisor depends on the OpenFlow version, which means that a new FlowVisor is needed for each OF version. Even more, private extensions could not be properly managed anyway. This deliverable introduces the Time-Based solution, which has one clear advantage: the control plane is transparently redirected to the adequate OpenFlow Controller. The main drawback is that the same resource cannot be shared at the same time by multiple experiments.

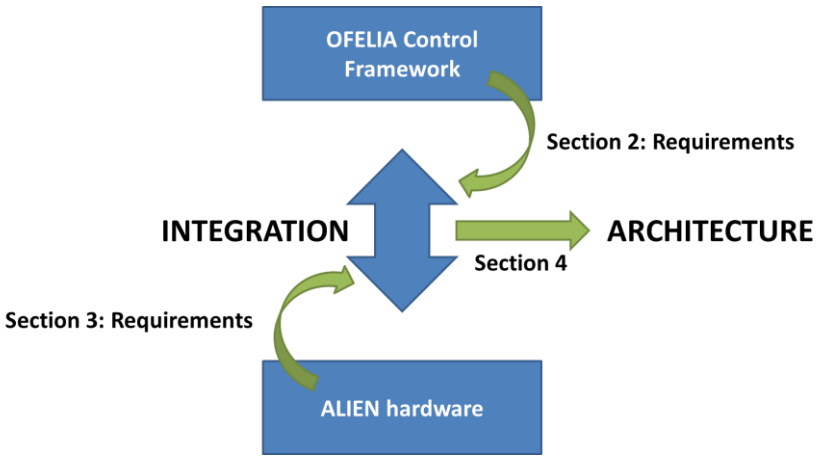

Figure 1.1: ALIEN hardware integration under OFELIA Control Framework

The document is organized as follows. First, Section 2 presents the OFELIA Control Framework and the requirements that it imposes to the integration of new resources. Then, Section 3 analyses the ALIEN hardware available in this project and the technical requirements each type of hardware provide. Furthermore, the integration plans for each platform are also presented. The preliminary architecture and description of the solution are detailed in Section 4. Finally, some conclusions are mentioned in Section 5.

2 Requirements for integration with the OFELIA Control Framework

The OFELIA Control Framework (OCF) can be defined as the control plane application for OFELIA -FP7 facility. The main purpose of the framework is to automate, simplify and authorize users to create network slices and deploy resources available within OFELIA islands for various types of experimental projects [OFELIA_D5.1].
To do so, the OCF provide a suite of software (Graphical User Interface, Aggregate Managers, Resource Managers) designated to manage the lifecycle of an experiment inside an OpenFlow environment and to automatically manage an heterogeneous set of resources (currently OpenFlow v1.0 flowspaces and XEN VMs). Such framework includes:

- User authentication
- Project creation and modification
- Slice creation and deletion, users and resources association
- Resource management, allocation and de-allocation
- Association between users, projects and resources

The reader is invited to leverage on the short glossary of OCF-related terms included in Appendix A in order to have a better understanding of the terminology used within this Chapter.

2.1 OFELIA Control Framework Architecture

The current OCF architecture has been designed based on GENI control framework and therefore it inherits its core components. However the architecture has been redesigned and extended extensively in order to make OCF more extensible. Also, a plug-in system in Expedient was introduced for better plug-in management. In addition, some new modules were added to the standard GENI architecture such as VT-AM, VT-Planner, VLAN-AM and FOAM. In the following, the architecture of OCF and its components are explained.

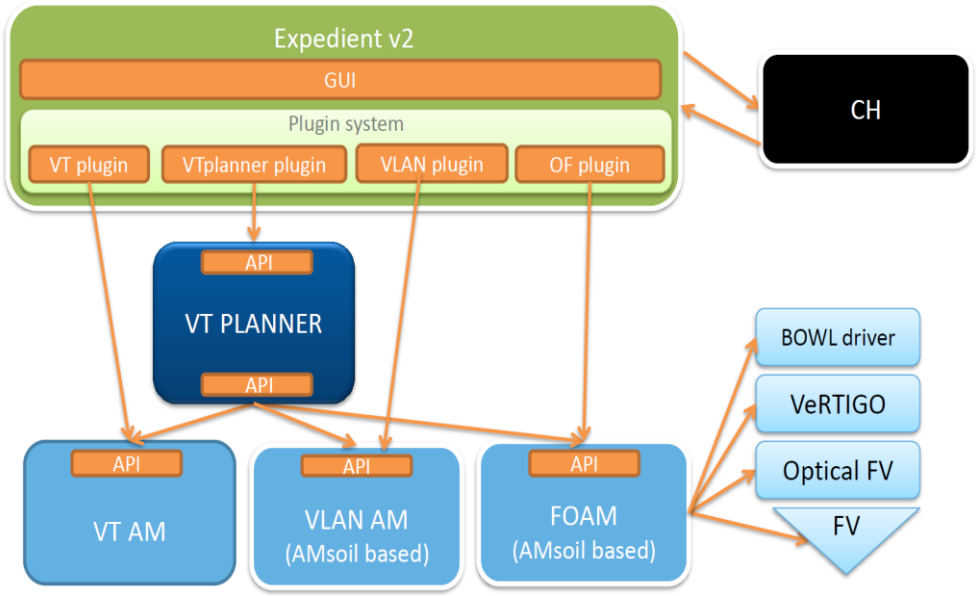

Figure 2.1: OCF architecture block diagram

2.1.1 Expedient v2

Expedient is a pluggable control framework with a central control block where different plug-ins can be connected to the correspondent AM. As mentioned before, each AM is responsible for the management of the underlying resources. These resources are presented to the user via Expedient user interface in a homogeneous way.

AM is the core element in the Expedient. Usually, all physical resources are controlled by some kind of management framework and interfaces (e.g. virtual machine as a monitoring tool for controlling the operation of virtual servers and the underlying physical environment). Each AM regarding of the resource's native management interface, can configure and monitor the resource. It also exposes the interface as an abstract interface to the Expedient control framework.

Since Expedient architecture is a modular approach, its functionality can be extended by adding new plug-ins and then AMs are connected to a resource specific plug-in within the Expedient. In other words, the AM binds resource's management interface to the Expedient via plug-ins in one-to-one relationship between AM and plug-in.

Apart from various plug-in that can be added to the Expedient, the tool itself consists of three main packages, a) Clearinghouse: a package which mainly deals with the information data to be stored in the database and managing certificates and credentials, b) Common: a package with auxiliary modules for messaging and XML-RPC services and c) UI: a package for managing user interface for controlling resources in a slice [OFELIA_D5.1].

2.1.2 AMSoil

AMsoil was designed as a base class to provide an environment to implement new AMs easily and dynamically. The AMSoil consists of two blocks: the resource specific part and the common part. The resource specific part deals with handling resources and the common part deals with general tasks needed by each AM such as identification, authentication, authorization, interface compliance and resource reservation management. In other words, it is a pluggable system, which provides logic or glue between RPC-Handlers and AMs [OFELIA_D5.2].

2.1.3 Virtual Technology Aggregate Manager (VT-AM)

The Virtual Technology Aggregate Manager [OFELIA_D5.1] enables users to create and control virtual machines on physical hosts. The VT-AM is connected northbound to the Virtualization plug-in in the Expedient and southbound to the OFELIA XEN Agent (OXA), that acts as Resource Manager. VT-AM gets required information for creating and controlling virtual machines from the Agent in the hosts and then gives this information to the Virtualization plug-in in the Expedient tool.

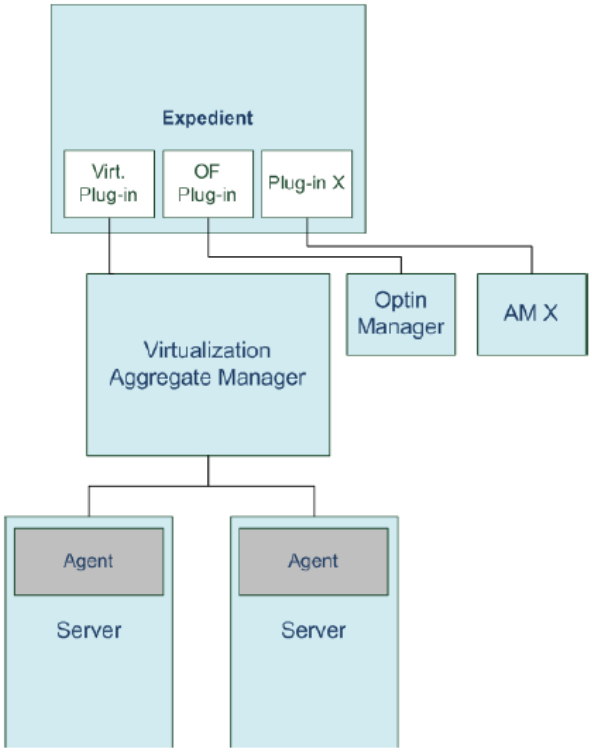

Figure 2.2: General Architecture of the VT-AM

The OXA resides in the host and it is responsible of managing the hypervisor installed on the host. Basically the Agent receives VT-AM messages from Northbound and applies them on the host's hypervisor via Xen scripts to satisfy user's requests. The main feature of the Agent is that it is agnostic to virtualization technology so that VT-AM can use the Agent on different hypervisor technologies.

2.1.4 Flowvisor OpenFlow Aggregate Manager (FOAM)

FOAM is the AM for handling OpenFlow resources through FlowVisor [FV] or VeRTIGO [OFELIA_VeRTIGO]. It allow the administrator to manage the entire flowspace and divide it through different slices (and different experiments), based on the users requests and the testbed policies. The Northbound API is used in OFELIA for communication with Expedient OpenFlow plug-in and the Southbound API is used for communication with FlowVisor or VeRTIGO [OFELIA_D5.1] [OFELIA_D5.2].
As FOAM is a GENI AM API-compliant Aggregate Manager, therefore it can be used to communicate with GENI's clients e.g OMNI. Also, it provides a mechanism to handle resources (create, approve, reject, disable, delete, list) which means administering multi-tenancy of the FlowVisor component to instantiate OpenFlow slices in the OFELIA context. Other

features of the FOAM are adding or deleting datapaths from testbed under control and providing pluggable administrative and configuration interface for OpenFlow network created in the OFELIA.

2.1.5 Virtual Topology Planner (VT Planner)

Within the OFELIA project some effort is dedicated to the design and implementation of new mechanism for the instantiation of virtual topology in OpenFlow networks, called VeRTIGO [OFELIA_VeRTIGO]. The main objective of VeRTIGO is the extension of FlowVisor in order to allow the experimenter not only to slice the network, but also to istantiate a virtual topology completely decoupled from the underlined physical network. To allow the interaction between the user and VeRTIGO, a new AM, called VT-planner, has been added to the OCF. VT Planner runs a resource allocation algorithm to embed requested virtual topology by the user into the underlying physical network and is connected northbound to the Expedient and southbound both to FOAM and to VT-AM. Resource allocation algorithm tries to accommodate the user requests to the current utilized system i.e. the current user's virtual topologies already allocated on top of the physical network and virtual resources associated with them [OFELIA_D8.2].

2.1.6 VLAN Aggregate Manager (VLAN AM)

Since VLANs are used as a slicing mechanism in the OFELIA facility, the VLAN AM aims to centralize the VLAN management in a single database.. The main functionality of VLAN AM is handling different VLANS over different OCF islands. The idea behind the VLAN AM is to make it perform as a DHCP server of VLANs. VLAN AM is based on AMsoil [OFELIA].

2.1.7 Clearinghouse

ClearingHouse (CH) serves as a trusted entity throughout the inter and intra-federation. The ClearingHouse is the only entity in the OCF architecture that deals with issuing user certificates and credentials. Each AM at least trusts one ClearingHouse, depending on the number of islands/organizations. The trust is issued via certificate chains in the system [OFELIA_D8.2].

2.2 ALIEN integration into OCF

Considering the modular architecture of OCF, which gives enough freedom to extend and customize the control framework, to introduce new entities (here ALIEN devices) into the OCF requires to implement new AM and new Plug-in in order to represent them to the user.
However, some requirements are inherited to the OCF, and need to be taken in account. The Table 2.1 below summarizes all the major requirements for the integration of ALIEN hardware within the OCF.

Requirement	Description
Dedicated device	Any device which is assigned to the OFELIA facility should be accessible all the time.
Shared resources	There should be a method to share the resources (or part of them) between experiments/researchers. Currently both FlowVisor and VeRTIGO are used to slice the OpenFlow flowspace on the physical OF switches (i.e. v1.0) among all the

	experiments and VT-Manager is used to virtualize the server infrastructure.
Backwards compatibility	The solution should be compatible (at least at data plane level) with the current OFELIA facility in order to allow inter-island experiments.
OpenFlow resources	To manage different OpenFlow version beyond v1.0, the OCF need to be upgraded. In particular, both FlowVisor (Resource Manager), FOAM (Aggregate Manager) and the Expedient OF Plugin (GUI plugin) are incompatible with other OF version except v1.0.

Table 2.1: Requirements for the integration of ALIEN into the OCF

3 ALIEN hardware and interface with the OFELIA Control Framework

The objective of this Section is to analyse the technical and organizational requirements imposed by the ALIEN hardware in order to be integrated under the OCF. Then, different approaches for resource integration are presented. Finally, it is explained how this integration of ALIEN hardware will be performed by each partner.

3.1 Technical and organizational requirements

All ALIEN hardware platforms for which Hardware Abstraction Layer (HAL) will be specified and developed have been elaborated in Deliverable D3.1 [ALIEN_D3.1]. In the summary of this deliverable five classes of the ALIEN hardware have been defined. Each of these device classes has different requirements for a potential integration with the Ofelia Control Framework (see Table 3.1).

Device type	Technical requirements/limitations
Packet processing devices	No additional requirements to OCF – the device can be managed as a standard OpenFlow device.
Programmable network processors	The programmability capabilities of the hardware should be exposed to the experimentation users. An exclusive access to the device should be provided in order to flash/reprogram the device. Access to the specific development environment (GUI application, SDK libraries) is required in order to program the device. The experimentation user may be interested in using different versions of the OpenFlow protocol.
Point to Multipoint Systems	In a basic setup no additional requirements are imposed to the OCF and the system can be managed similarly to standard OpenFlow devices. The distributed nature of this type of systems (initial connection must be established) and the asymmetry between the headend (i.e. OLT and CMTS) and the leaves (i.e. ONUs and cablemodems) should be considered. The virtualization (slicing) using VLAN tags could be not available in configurations in which VLANs are used internally to distinguish

	clients' traffic.
Lightpath Systems	Device's data plane interfaces are purely optical and therefore require different information modeling and representation within the OCF. For example, the virtualization (slicing) can be done using lambdas instead of VLANs.
Physically Reconfigurable Systems	A single device may be modeled as few network nodes, each of them controlled independently using OpenFlow. Some components of the system could have special capabilities which could be manageable from the OCF framework.

Table 3.1: Technical requirements to integration with OCF for given types of the ALIEN hardware

The Table 3.1 shows the dependence of the technical requirements with the specific types of a device. This means that each ALIEN hardware imposes a different set of requirements, making even more complex the design of a general/common solution for their integration. The idea of a general approach to the integration of the ALIEN hardware is referred to as the ALIEN AM (ALIEN Aggregate Manager), which is further detailed in Section 4.

Besides the technical requirements for the integration with the OCF, partners may have some organizational restrictions/limitations which will directly impact the final decision how each ALIEN device will be integrated within the experimentation network, and how the OCF framework will be extended in order to manage the experimentation resources from the ALIEN project. The organizational implications for the ALIEN hardware integration are presented on Table 3.2.

Partner	Organizational implications
PSNC	EZappliance is currently managed from PL-LAB (which is an alternative solution to the OCF). The possible solutions for sharing EZappliance between PL-LAB and ALIEN will be investigated.
UPV/EHU	DOCSIS is used during academic lessons so exclusive access for the OCF users is not possible.
Create-Net	No special organizational implications for the integration with the OCF.
PUT	No special organizational implications for integration with the OCF. However, PUT is not a member of the ALIEN AM integration task.
UCL	Because UCL is not member of the ALIEN AM integration task, it has been decided UCL will not act as an OFELIA island.
UNIBRISTOL	No special organizational implications for the integration with the OCF.
EICT	No special organizational implications for the integration with the OCF.
DELL	No special organizational implications for the integration with the OCF, however, DELL is not a member of the ALIEN AM integration task.

Table 3.2: Organizational implications to the hardware integration with OCF

Although there are some organizational limitations that could imply in the future some restrictions, the integration of the ALIEN hardware under the OCF is proposed in such a way that all the platforms will be covered. This means that if certain partners do not have efforts for the integration task, their platforms are not left out from the solution. However, the development and validation activities will focus on those platforms from partners with efforts allocated to the integration activities.

3.2 Resource integration approaches

After the analysis of technical requirements for each ALIEN hardware and also partners' organizational implications, three alternative approaches for ALIEN hardware integration have been defined. The first approach is "OCF Standard" which is the default approach. The next approach is "No OCF" which is defined for resources that cannot be under management of the OCF framework because of the organizational implications (Table 3.2). The third integration solution, the "ALIEN AM", is designed to address other requirements such as the ones defined for programmable network processors (see Table 3.1). More details for each integration approach are presented in Table 3.3.

Resource integration approach	OpenFlow version supported	Integration plane(s)	Description
OCF Standard	OFv1.0, circuit extensions v0.3 available	Management Plane, Control Plane and Data Plane	The integration is done following the OFELIA project procedures. The usage of OpenFlow v1.0 is mandatory. Experiments are distinguished within Data Plane based on VLAN tags. The main benefit of this approach is that there are no additional development efforts needed.
ALIEN AM	OF version agnostic	Management Plane, Control Plane and Data Plane	The integration with the OCF is established with a new Aggregation Manager developed within the ALIEN project. The ALIEN AM is a general concept which covers the necessity of developing a new component to integrate the ALIEN hardware under the OCF. Section 4 introduces a particular solution / implementation for the ALIEN AM which is called the Time-Based Aggregate Manager (TBAM). The TBAM will allow time-based sharing of resources within OCF framework. A single user at each time can have the complete control of the hardware (for a fixed amount of time). This approach is preferable in order to expose any specific hardware capabilities (e.g.: programmability) to experimentation users and also to allow experimentation with newer OpenFlow versions (v1.2, v1.3, …). The implication of this approach is that an access to the management interface of the device must be provided and some development efforts will be required in order to integrate the ALIEN hardware with OCF.
No OCF	An experiment user independent decision	Control Plane and Data Plane	The resources are not under the control of the OCF. On one hand, an integration task with other resources focuses mainly on the Data Plane, since it should be coherent and consistent at experiment level even with the legacy resources currently available at OFELIA. On the other hand, the Control Plane

	actions will be redirected to the appropriate controller. This approach gives much less constrains for the integration of resources to an experiment.

Table 3.3: Resource integration approaches for ALIEN hardware

In a nutshell, the objective of this document is to deal with the "ALIEN AM" approach for the resource integration. In fact, the "OCF Standard" approach implies that those platforms which select this alternative are already able to be integrated with the current OCF. Moreover, the "No OCF" option means that no effort will be done to try to integrate that platform under the OCF (different reasons are explained in Table 3.2). Consequently, this project focuses on the design and implementation of the "ALIEN AM". It should be general enough to cover all the platforms and possible OpenFlow versions. In Section 4 a concrete proposal for implementing such an approach is detailed, the Time-Based Aggregate Manager.

3.3 ALIEN hardware integration

Each partner has chosen the integration approach for the ALIEN hardware and which OpenFlow version would like to expose for experiments and demonstrations (see Table 3.4). In case of the "ALIEN AM" approach, the partner was asked to describe the management interface(s) of their ALIEN hardware that could be used to manage the resource by OCF control. In case of "No OCF" approach, and in order to show how this ALIEN device can be integrated into the final testbed in WP5, both the Control and Data planes connection should be specified. This connection could be provided by direct links (i.e. Layer 1 connection) between the ALIEN device and some OFELIA island. One important aspect with regard to the Data Plane is the possibility of supporting VLAN-based slicing to check the compatibility with legacy devices which are available at OFELIA.

Partner	Hardware	OF version	Integration approach	Management interface	Control/Data Plane access point
PSNC	EZappliance	OFv1.0, OFv1.3	ALIEN AM	Corba / HTTP / console / EZdriver API	N/A
UPV/EHU	DOCSIS	OFv1.0	No OCF	N/A	Layer1 connection to EHU island
	ATCA	OFv1.0, OFv1.2, OFv1.3	ALIEN AM	Ssh	N/A
PUT	EZappliance	OFv1.0	ALIEN AM	ssh / webconsole	N/A
	NetFPGA	OFv1.0	ALIEN AM	ssh / telnet / webconsole	N/A
UNIBRIS	L0 switch	OFv1.0 with circuit ext v0.3	OCF Standard	TL1 port (2024) via telnet, ssh, web GUI, SNMP	N/A
UCL	GEPON	OFv1.0, OFv1.3	No OCF	N/A	Layer 1 connection to UCL tesbed (not OFELIA island)
DELL	Dell switch PC|7024 with a NPU Octeon	OF 1.0 and OF1.1+ in dev	No OCF	N/A	Layer1 connection to Dell island

	CN52xx				
EICT	ATCA	OFv1.0, OFv1.2, OFv1.3	ALIEN AM	ssh	N/A

Table 3.4: Partners' preferences on the ALIEN hardware integration

As previously mentioned, this deliverable deals with those resources which are going to implement and test the "ALIEN AM". This means that those resources with the "No OCF" or "OCF Standard" approaches are out of scope of this document. Mainly, there are two reasons that justify the necessity of the "ALIEN AM" in order to integrate the ALIEN hardware under the OCF: implementing the version of OpenFlow beyond 1.0 and the requirement of exclusive access to the resource. The former invalidates the use of FlowVisor (at least in its current version, which supports OFv1.0 only) to share the resource among experiments/researchers by the OCF. The latter only permits one researcher at a time accessing the resource, and therefore, sharing it at the same time is not possible. The Time-Based Aggregate Manager detailed in Section 4, is a concrete proposal of the "ALIEN AM" implementation tackling with both abovementioned issues.

4 OCF extensions preliminary architecture

The objective of this Section is to provide a preliminary definition of the extensions to be implemented within the OFELIA Control Framework in order to allow ALIEN hardware to be accessed and shared by many researchers willing to perform experimentation on it on an OFELIA island.

4.1 Introduction

The focus of WP4 is to integrate the ALIEN resources inside the OFELIA Control Framework (OCF), in order to allow an experimenter to seamlessly access to both "standard" OFv1.0 devices and ALIEN Hardware, and of course to share these resources among all OFELIA users. For this reason, the OCF has to be extended to support the new features deployed within ALIEN project, in particular the ability of the ALIEN hardware to go beyond the OpenFlow v1.0.

As mentioned in Section 2, the current slicing mechanism for OF switches is either FlowVisor or VeRTIGO, which are managed by the FOAM AM [FOAM] inside the OCF. Two main motivations imposes an extension to the current version of the OCF to enable ALIEN hardware management:

- The slicing mechanism used in OFELIA is tightly linked to the OpenFlow version. Currently there is no available software for slicing the control plane of OpenFlow resources running OF versions beyond v1.0, and the roadmap for the implementation of this functionality inside FlowVisor (or VeRTIGO) is not clearly defined yet (see also Section 4.3). As mentioned inside the deliverable D2.1, the HAL will be not strictly linked to the OFv1.0, but will probably move toward OFv1.3, to expose a superset of features offered by the ALIEN hardware.
- The hardware (OF switches and servers) within OFELIA testbed is shared on a "best-effort" basis; in other words, when multiple users are sharing the same resources at the same time, there are no mechanisms to avoid congestions or limiting the amount of resources assigned to the specific experiment (e.g. CPU, memory, bandwidth, ...). This choice, primarily due to the fact that the only mechanisms for sharing resources inside the OpenFlow domain was (and is) FlowVisor, prevent the full access of the hardware itself, both to install the specific user software and to conduct some performance tests that require all the resources or a set of them, in order to permit the repeatability of the analysis for example.

In order to overcome the current limitations of OCF, it has been envisioned to implement a "time-based" sharing mechanism in order to share ALIEN hardware among experimenters within an OFELIA island. This solution can be integrated inside the OCF and can coexist with the standard OpenFlow Aggregate Manager as an extension, not as its replacement (thus being backward compatible with the current OCF running on existing OFELIA islands, letting each and every island administrator to decide if they want to enable or not this specific extension into their testbed).

4.2 Time-Based Aggregate Manager (TBAM)

The idea behind the TBAM is both to go beyond the current limitation of the OCF and to allow the integration of the ALIEN hardware inside it. Figure 4.1 represents the current OCF and the proposed modules (in red) to extend it in order to implement the Time-based AM.

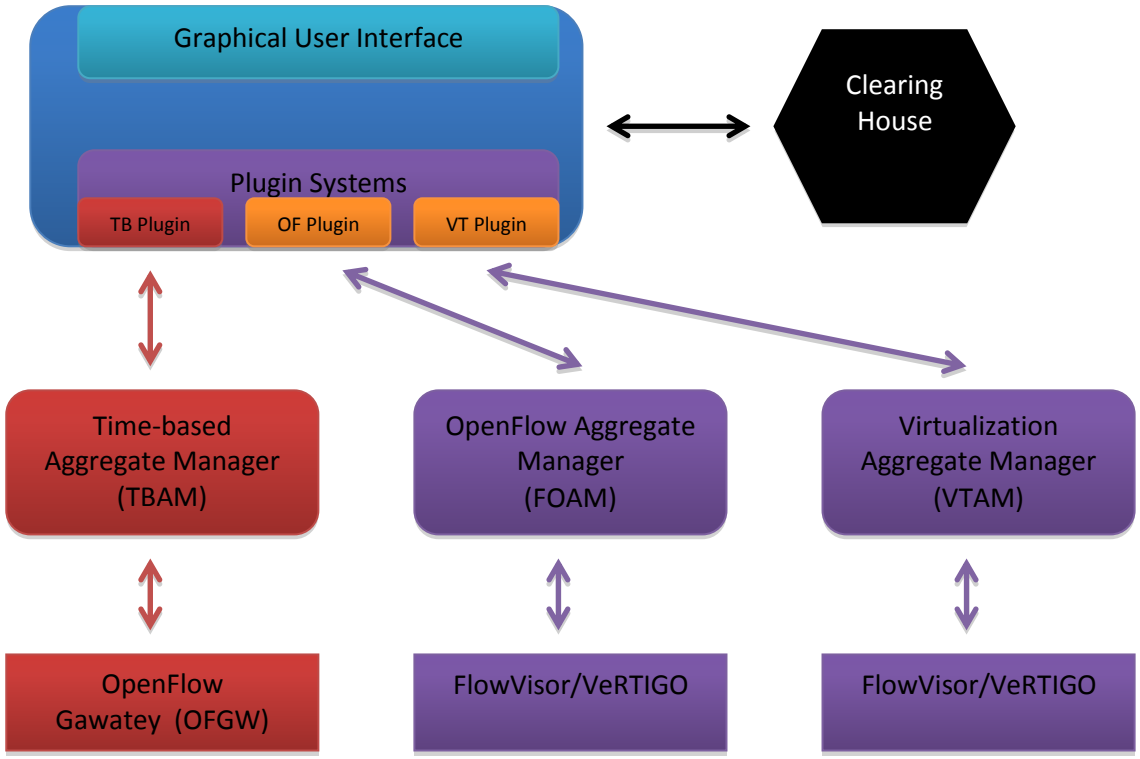

Figure 4.1: OFELIA Control Framework including the Time-based AM

In the following, we will provide a preliminary description of the software modules that need to be implemented (or extended) in order to realize such TBAM:

- Graphical User Interface Plugin (GUI - **extension**): The current OCF GUI (implemented in Django [Python_Django]) has to be extended to support the new TBAM, both at the user frontend and in the plugin system. In other words, a specific plugin must be implemented to connect the TBAM to the GUI; of course, the APIs between all the new modules need to be properly defined. Ideally, the interface used could be GENI AM API v3 [GENI_API3], already adopted inside the OCF. Moreover, the GUI has to be extended to include a specific module to manage the resources exposed by the TBAM. In particular, this web form will allow the user to:
  - Book a specific subset of ALIEN resources for a fixed amount of time
  - Release the resources either automatically at the end of the time interval, or manually if the user wants to leave the project sooner
  - Modify both the amount of resources booked and the time interval of the project
  - Inform (via email) the user and the ALIEN resources manager about all the relevant operations executed with their related status

- Manage concurrent requests (e.g.: inform the user that another user is interested in the same resources so he/she can release them if unused,…)

- Time-based Aggregate Manager (TBAM – **new**): The AM will expose to the GUI all the diverse "time-based" resources coming from different Resource Managers (named OFGW, see next bullet). As mentioned in Appendix A, each AM is responsible of the management of an homogeneous set of specific elements (such as switches, servers,…). In this scenario, the TBAM will maintain the status of the resources, the association between users/projects and booked elements, and all the information related to the specific management.

- Resource Manager: OpenFlow Gateway (OFGW – **new**): The OFGW will be the only entry point to access the "time-based" resources under its control. It will act both as specific manager for the resources and as gateway for the data and control network.

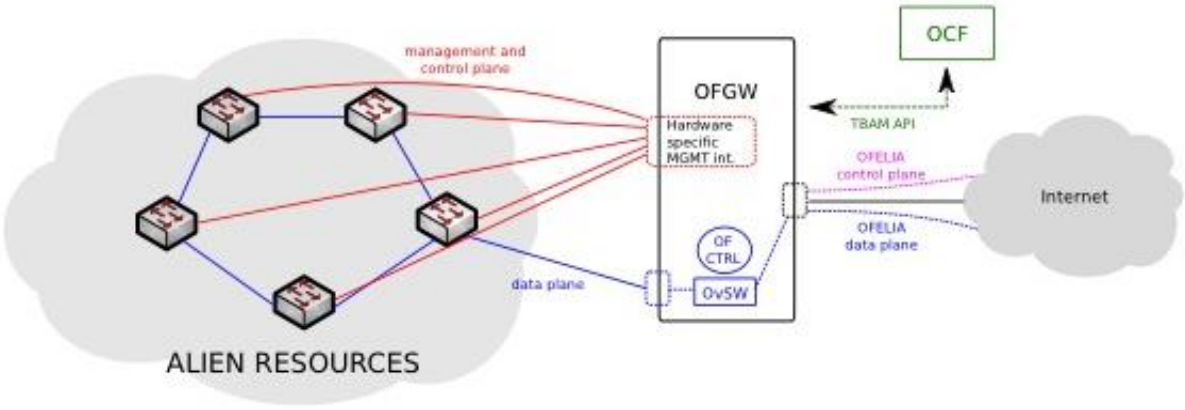

Figure 4.2: OFGW architecture and interfaces

This Resource Manager is in charge of:

- *Hardware Management Plane*: as depicted in Figure 4.2 (red arrows), the OFGW has "administrator" credentials to access all the resources through hardware specific interfaces (e.g. console, telnet, ssh,… see Section 3);

- *Control Plane*: the OFGW will act as the unique entry point for the OpenFlow control plane. For this reason, it will be configured with an IP address within the OFELIA control network, and it will gain the role of proxy of the control messages between the OF switches and the controller. This will avoid any possible problems in the routing of packets from the internal network to the OFELIA facility.

- *Data Plane*: while inside "canonical" OFELIA islands, the slicing mechanism used is based on VLAN tagging, within ALIEN we assume that the traffic inside the "time-based" resources is, by default, untagged (VLAN 1), given that the user is willing to have full control of the hardware for the time needed to experiment with it. However, also tagged traffic will be allowed. As traffic inside OFELIA island must be tagged with the VLAN assigned to the slice, the packets exchanged between OFELIA and ALIEN nodes must go through a tagging/untagging process performed by the OFGW. For this reason, the OFGW will be equipped with an OpenFlow capable software switch (e.g. OpenVSwitch [OVS]) and

an open-source OF Controller (e.g. Floodlight [Floodlight] or Beacon [Beacon]), directly managed by the control framework.

To summarize the VLAN management, the traffic of ALIEN resources is assumed to be untagged by default, and in case of a multiple island experiment, an unique VLAN per experiment will be allowed too. However, multiple VLANs are also allowed for specific experiments. If necessary, it will possible to map the inner VLANs (ALIEN resources) to the outer VLANs (OFELIA facility) through the GUI (TB plugin), and the mapping will be automatically added in the OF Controller that will be in charge of the rewriting.

Finally, different version of OpenFlow protocols could be simultaneously available within the same experiment (e.g. OFv1.3 ALIEN hardware + OFv1.0 in OFELIA island). It is not strictly related to the work of the WP4 to take care of possible incompatibilities between controllers and switches (e.g. it could be necessary to define two different controllers to overcome such problem).

4.3 Future Investigations

It is important to highlight that novel solutions to virtualize OpenFlow and ALIEN resources will be investigated within WP2 (in particular, in Task 2.4 "*Reservation and virtualization of resources*"). While the discussion within that task is still at an early stage, two approaches seem interesting to be investigated: (i) to include a virtualization layer directly on top of the HAL, in order to provide a distributed and completely autonomous virtualization framework, and (ii) to extend FlowVisor which is currently limited to work only with network resources running OFv1.0. ON.Lab, a branch of the ONRC (Open Networking Research Center, a joint research facility between Stanford and Berkeley universities), is in charge of FlowVisor deployment and is currently working on a new approach for the OF virtualization which should be less dependent on the different OF versions and thus more suitable for virtualizing some of the ALIEN resources.

The on-going activities within Task 2.4 could have some impact on WP4, in terms of further extensions to the OFELIA Control Framework. However, at least for the time being, the core work within WP4 will be the design and implementation of the "Time-Based" virtualization approach described above.

5 Conclusions

This deliverable presents all the requirements that should be considered to properly design the integration of ALIEN hardware under the OFELIA Control Framework. On one hand, the OCF is oriented to manage the life-cycle of the experiments which share the same infrastructure and resources, i.e. the experimental facility. This control framework imposes just a few requirements, because it was designed to be flexible (OCF is based on Expedient), modular and to support very different types of resources. On the other hand, this project deals with very different types of ALIEN hardware. In fact, it was conceived to cover a huge variety of devices with very different characteristics. Each type of hardware imposes different requirements to the final solution and its integration under the OCF. Moreover, each partner has different objectives and plans for its own hardware (support for v1.0, v1.3, and so on).

With all those requirements taken into consideration, this deliverable has tried to find an overall solution to cover all the possible ALIEN hardware available in WP3 in the most general manner. The ALIEN hardware deals with different versions of OpenFlow (i.e. v1.0, optical extensions v0.3, v1.1, v1.2, v1.3 or private extensions to one of those standard versions), so the solution was designed to support all these options. Consequently, this approach does not inspect the control protocol (i.e. OpenFlow) and redirects the whole control plane to the adequate Controller from the proper researcher or experiment. The sharing of resources is a requirement imposed by the OCF, which is achieved by implementing the Time-Based sharing.

It must be said, that for different reasons, all the hardware available in WP3 will not be integrated in this WP4 behind the OCF, although the solution has been designed to cover all type of hardware.

Another aspect that should be highlighted is that the compatibility of this solution with the legacy OFELIA resources imposes that both approaches should be compatible at data plane. Since the legacy OCF has defined the VLAN tag as the mechanism to slice the legacy resources, the new approach should adapt the data plane of the ALIEN hardware before accessing the legacy resources. However, this requirement is not new for OFELIA. The OFELIA project developed the integration of the VMs under the OCF, which is based on traditional virtualization solutions (i.e. Xen). In this case, the VMs should be adapted at data plane before accessing the networking resources (i.e. the OpenFlow switches). Each researcher needs to configure the interfaces of his/her VM with the appropriate VLAN tag. The configuration of this VLAN is essential (and a requirement) to be compatible at data plane with the rest of the resources assigned to the experiment.

The proposed solution tries to be general enough to avoid future modifications. This is the reason why the OpenFlow protocol is not inspected. However, it has some limitations. For instance, the resources cannot be shared at the same

time by multiple experiments/researchers. This could be an issue, but in some cases, the ALIEN itself needs a specific development environment that does not support concurrent access (i.e. development applications and SDK with ends in flashing the hardware).

One of the positive consequences of the Time-Based sharing is that it enables a researcher to get access to the full capacity of the resource (not being affected by other experiments).

To conclude, it must be said that the Time-Based sharing can be an alternative to integrate other types of resources under the OFELIA Control Framework. This approach introduces some unique characteristics that could be beneficial in some scenarios.

The ALIEN project will evaluate the possibility of integrating the ALIEN AM (i.e. TBAM) into the main open-source OCF development framework in order to be available to all current (and future) OFELIA islands or any other project/deployment which makes use of this OCF (e.g. GEANT OpenFlow-based testbed).

6 References

[ALIEN_D2.1]	Deliverable 2.1 “Report on Hardware abstraction models”, FP7-ALIEN, 01-2013
[ALIEN_D3.1]	Deliverable 3.1 “Hardware platforms and switching constraints”, FP7-ALIEN, 02-2013
[ALIEN_HAL]	HAL definition in Ch.2 of ALIEN D2.1 “Report on Hardware abstraction models”, FP7-ALIEN, 01-2013
[Beacon]	Beacon Web-site, https://openflow.stanford.edu/display/Beacon/Home
[Floodlight]	Floodlight Web-site, http://www.projectfloodlight.org/floodlight/
[FOAM]	FOAM Website, https://openflow.stanford.edu/display/FOAM/Home
[FV]	FlowVisor Web-site, https://openflow.stanford.edu/display/DOCS/Flowvisor
[GENI_API3]	GENI AM API v3 Web-Site, http://groups.geni.net/geni/wiki/GAPI_AM_API_V3
[OCF]	OFELIA Control Framework development wiki [online]. https://github.com/fp7-ofelia/ocf/wiki/Development
[OFELIA]	OFELIA documentation wiki. [online]. https://alpha.fp7-ofelia.eu/doc/index.php/Main_Page
[OFELIA_D5.1]	OFELIA deliverable 5.1. [online]. http://www.fp7-ofelia.eu/assets/Public-Deliverables/OFELIAD5-1-final.pdf
[OFELIA_D5.2]	OFELIA deliverable 5.2. [online]. http://www.fp7-ofelia.eu/assets/Public-Deliverables/OFELIAD5.2Second-version-of-OFELIA-Management-Softwarev2.pdf
[OFELIA_D8.2]	OFELIA deliverable 8.2. [online]. http://www.fp7-ofelia.eu/assets/Public-Deliverables/OFELIAD8.2-final.pdf
[OFELIA_VeRTIGO]	VeRTIGO Web-site, https://github.com/fp7-ofelia/VeRTIGO/wiki
[OVS]	OpenVSwitch Web-Site, http://openvswitch.org
[Python_Django]	Django Web-site, https://www.djangoproject.com
[Xen]	Xen project [online]. http://www.xen.org/

7 Acronyms

[AM]	Aggregate Manager
[CH]	ClearingHouse
[FOAM]	Flowvisor OpenFlow Aggregate Manager
[GUI]	Graphical User Interface
[GW]	Gateway
[HAL]	Hardware Abstraction Layer
[OCF]	OFELIA Control Framework
[OF]	OpenFlow
[OFGW]	OpenFlow Gateway Resource Manager
[OVS]	OpenVSwitch
[OXA]	OFELIA XEN Agent
[RM]	Resource Manager
[TBAM]	Time-Based Aggregate Manager
[UI]	User Interface
[VLAN AM]	VLAN Aggregate Manager
[VL]	Virtualization Layer
[VM]	Virtual Machine
[VT-AM]	Virtual Technology Aggregate Manager
[VT Planner]	Virtual Topology Planner

8 Appendix A – OCF glossary of terms

Below, a list of terms and definitions are introduced to facilitate the understanding of the OFELIA Control Framework [OCF]:

- **Administrator**: An administrator is a user in OFELIA facility and/or OFELIA island management team, who is responsible for installing, maintaining and supporting the resources and processes.
- **Aggregate**: A composition of resources is an aggregate.
- **Aggregate Manager**: An Aggregate Manager is an entity, which manages multiple underlying aggregate(s) or resource managers.
- **Component**: A component is a physical resource.
- **Experimenter/Researcher**: A researcher is a user who utilizes the OFELIA facility to conduct experiments via the OFELIA control framework.
- **Inter-Federation**: The inter-federation is the experimental environment created by inter-linking multiple experimental testbeds managed by different control frameworks and different administrative domains.
- **Intra-Federation**: The intra-federation is the experimental environment created by the individual OFELIA islands with heterogeneous experimental facilities using the homogeneous control framework to appear as a unified experimental facility to the end-user.
- **OFELIA Facility**: The OFELIA facility is an experimental facility which offers an infrastructure to conduct OpenFlow-enabled experiments.
- **OFELIA Island**: An OFELIA island is an autonomous system which has its own administrative domain offering its experimental testbed with the OpenFlow-enabled network components and end systems to the OFELIA facility. An OFELIA island is part of the OFELIA facility.
- **Project**: A project contains users, experiments and slices.
- **Resource**: A resource is a component or a subset of a component that can be booked.
- **Resource manager**: Resource manager is an entity, which manages the resource and maintains the resource states.
- **Resource state**: A resource could be either in "available" or "reserved" or "instantiated" state. The "available" state means that the resource is available to be reserved for a given period of time. The "reserved" state indicates that the resource is reserved for a given time frame and the "instantiated" state refers to the state, in which the resource is actually engaged in an experiment (run, stopped, paused, etc.).
- **Slice**: Slice is a set of reserved resources.
- **User**: A user is a person who uses the OFELIA facility via the OFELIA control framework.