OConS: Towards Open Connectivity Services in the Future Internet

Ramón Agüero[1], Luisa Caeiro[2], Luís M. Correia[2], Lúcio S. Ferreira[2],
Marta García-Arranz[1], Lucian Suciu[3], and Andreas Timm-Giel[4]

[1] University of Cantabria, Santander, Spain
`{ramon,marta}@tlmat.unican.es`
[2] IST/IT Technical University of Lisbon, Lisbon, Portugal
`{luis.correia,lucio.ferreira}@lx.it.pt`, `lcaeiro@est.ips.pt`
[3] Orange-Labs, Rennes, France
`lucian.suciu@orange-ftgroup.com`
[4] Hamburg University of Technology, Germany
`timm-giel@tuhh.de`

Abstract. The recent advances on networking technologies (both at the access and the core realms) together with the ever-increasing requirements of the end-users and their applications/services call for an open approach, yet with a clear migration strategy, so as to avoid the well-known shortcomings and limitations of clean-slate approaches. These requirements have streamlined the design of a novel (yet not revolutionary) architecture framework based on the identification of functional entities and their interfaces. The most distinguishing feature is its flexibility, allowing its adaptation to already existing protocols/technologies/algorithms as well as to novel solutions.

Keywords: Future Internet, Open Connectivity Services, Design Guidelines.

1 Introduction and Related Work

It goes without saying that the motto *'Internet of the Future'* has recently attracted the interest of the scientific community and, therefore, different proposals have been made so as to tackle some of the identified challenges and characteristics. They cover a broad scope and they also have a wide range of different characteristics and philosophies, i.e., revolutionary proposals pursuing a clean slate approach, while others foster a migration strategy for legacy architectures.

This paper presents the architecture which has been adopted for managing open connectivity services in the framework of the *Scalable and Adaptive Internet Solutions* (SAIL) Project [14]. The OConS framework aims at tackling some of the most relevant challenges which are posed by new communication paradigms, brought about by the so-called Internet of the Future. It is now believed that traditional networking approaches are not appropriate anymore, and in addition, patches and evolutions of currently available architectures are

K. Pentikousis et al. (Eds.): MONAMI 2011, LNICST 97, pp. 90–104, 2012.

deemed insufficient. Therefore, novel architectures have been proposed in the recent years, some of them even being clean-slate approaches. However, a clean-slate approach may also come with disadvantages, such as the need for valid migration strategies to ensure a relatively quick rollout and deployment, and the risk of improving some aspect of the network while creating unforeseen new problems. It is therefore widely recommended not to dismiss everything, but rather to build on what is working well, only replacing or ameliorating the unsatisfactory mechanisms or protocols. OConS aims thus at addressing the challenges which characterize the upcoming communication environments, while providing a sound migration strategy. The way forward is therefore to foster an open environment, flexible enough to accommodate most of the currently available procedures and to suit the needs for the forthcoming ones. This openness is the most distinctive feature (and requirement) of the OConS approach, yet we need to tackle some additional aspects which are briefly introduced below. As it has been briefly mentioned above, the architecture should be able to adapt to the rapid evolution of the communication technologies and related processes. This flexibility shall also span to the dynamic creation of networking and connectivity services in an autonomous manner (i.e., self-configuration, self-optimisation, self-healing, self-management); if possible, this creation should be based on the activation and de-activation of the already existing modules and mechanisms. Last, but not least, and from a general perspective, it is also of outer relevance to highlight the need of a distributed/collaborative architecture for control and management. Centralised approaches, albeit reducing the inherent system complexity, might lead to scalability and robustness issues and, therefore, considering the rapid growth of nodes might become unacceptable. A direct consequence of this distributed approach is that the system should provide the means to discover the available connectivity services.

A key distinguishing feature of OConS is its holistic approach; in this sense, as opposed to the various proposals and works which are cited hereinafter, OConS will provide a common wrapper so as to ease the process of integrating different techniques, protocols, and algorithms, ranging for the access part of the network to the interconnection of data-centres, while facilitating the interoperation between them. Nonetheless, some of the most relevant activities in the related areas have streamlined parts of the design of the OConS architecture and it is worth enumerating them.

Starting with the heterogeneous access networks, two lines of work can be highlighted here. The first one corresponds to the work carried out (mostly) by different research initiatives which aimed at proposing novel architectures to deal with the access selection in heterogeneous scenarios. In this realm, one of the most relevant proposals was the EU Ambient Networks Project, see [10] and the references therein. This project designed a networking architecture, aiming at leveraging the cooperation between different networks, embracing mobility, context-awareness, security, and other control functions[1]

---

[1] Although Ambient Networks also aimed at looking at the end-to-end connectivity, the focus was given to the access part [11].

The second one concerns the efforts taken by the relevant standardisation bodies. For instance, the 3GPP Evolved Packet System (EPS) supports both the existing accesses (i.e., 2G/3G) as well as the interworking between 3GPP and non-3GPP alternatives (e.g. Wi-Fi). Besides, the IEEE 802.21 Media Independent Handover (MIH) standard [6], defines media access independent mechanisms aiming at enabling seamless handovers between IEEE 802 (802.11, 802.16, or 802.3) systems and non-IEEE 802 (e.g. 3GPP, 3GPP2) cellular systems. The OConS will go beyond the current scope of the MIH framework, since it will also consider limitations, policies, rules and requirements coming from other parts of the network, rather than from only the subjacent link layer technologies.

Regarding the core networking techniques, there are various lines of research which are worth looking at. The Generalised Multiprotocol Label Switching (GMPLS) is intended to bridge the gap between the lower layer (e.g. optical) transport infrastructure and the IP layer. It is also designed to enable multi-vendor interoperability and multilayer functionality [8]. Besides, IETF TRILL standard, or IEEE 802.1aq Shortest Path Bridging, provides a method of interconnecting links that combines the advantages of bridging and routing [12].

The Path Computation Element (PCE) architecture [2] has been introduced to provide effective Traffic Engineering solutions. The main motivations that drove the introduction of the PCE architecture included the need to perform CPU-intensive path computations and to deal with several scenarios where the node responsible for path computation has limited visibility of the network topology and resources (e.g., multi-domain and multi-layer networks).

Finally, it could be highlighted the efforts on the multipath transport protocols realm. This has become an extensive and diversified research area, with various proposals ranging from modifying the currently prevalent protocol TCP [5] to proposing generic transport for the Future Internet [3]; moreover, they can be applied to different layers or entities, such as routing and transport protocols, applications (e.g. in a peer to peer overlay) or anywhere in-between.

The OpenFlow novel concept deserves some particular considerations. The discussion above leads to a clear conclusion: a large number of networking concepts have recently flourished. Starting from the observation that these newly conceived networking concepts can barely be deployed and tested, the OpenFlow [9] framework has recently taken roots. Thus, its main goal is to make networks programmable by manipulation of the entries of the flow table, e.g., in an Ethernet switch via an open interface implemented by the OpenFlow protocol, making possible to control the network traffic more easily.

Lastly, the 4WARD project [4, 13, 15] developed a clean-slate architectural framework based on the Generic Path (GP) concept. The main objective of the GP model was the support of various communication needs in highly mobile and dynamic networking conditions, while adapting the end-to-end transport and QoS procedures to the capabilities of the underlying networks. In addition, it also benefits from paths diversity over multiple routes as well as the introduction of advanced techniques such as network coding.

As has been seen, and without having been exhaustive, there is a large number of different proposals that OConS should integrate into a common framework, so as to instantiate the appropriate technique depending on the particular needs of the end-user/service/operator. The flexibility and the openness pose major difficulties to be overcome, these becoming even more relevant if we consider the need for a clear migration strategy, which implies that the deployment and roll-out timing should be realistic and as short as possible.

The present paper is organized as follows. Section 2 identifies key requirements for the OConS architecture. Design guidelines for OConS are presented in Section 3, while the architectural framework is detailed in Section 4. In Section 5 conclusions are drawn and the plans for future work are given.

2 Requirements for an OConS Architecture

As mentioned before, OConS addresses the challenges which have brought about by new communication paradigms. In this section, the specific challenges of the technical areas OConS particularly deals with, namely routing, transport, security, mobility and resource management, are discussed. The identified requirements serve as the basis to propose the barebones of the OConS architecture, as described in Section 4.

Requirements on Routing: The OConS mechanisms must address general expected requirements linked to other globally desired features such as: (i) the suitability of strategies even under conditions of mobility, (ii) the consideration of security as a primary concern within the design phase of the strategies, and (iii) the concept of multi-path as the norm rather than the exception when deciding the routing strategy. Routing is supposed to tackle the general needs assumed by routing in heterogeneous environments. End-to-end routing shall be provided across heterogeneous physical technologies such as optical, wireless, or copper based networks. It shall be implemented using a multi-domain paradigm (administrative, policy or trust domains), with domains capable to exchange comparable tokens of information in a secure way. Innovative topologies and deployments, such as challenged networks, demand new routing approaches able to self-adapt to changing conditions. Effective communications among entities on different layers shall be supported through cross-layer signalling.

Requirements on Transport: These encompass support for a wide range of flexible solutions to enable efficient and optimised services. One of those is the support for multiple paths, within the novel requirements of edge-to-edge, where the transport services are delivered between the network edges. Such delineators may be defined as any set of end points (locators), which may be associated by a multi-homed device or by a network cloud, i.e., a set of multi-domain end points. This also concerns the support of challenged-networks, where a pair of nodes needing some flow control to regulate the exchange of data from one node to the next, and this can be regarded as a pair of “edges” in the sense described above. Optimised multi-path transport is also required to support applications with heterogeneous content, by enabling customisable transport parameters (e.g., congestion control type, reliability and in-order delivery) within selected paths. Fair and efficient congestion control algorithms that use all (or some) of the available paths are also required, so as to increase the throughput without hurting concurrent, legacy flows.

Requirements on Security: The OConS framework should not only address suitable characteristics for resilience, scalability and manageability. It should also ensure that it cannot be misused such that the system integrity is endangered. Requirements regarding security are identified as security objectives to describe protection targets according to some security policy. Security objectives are the legitimate use of the advanced mobility management, preventing misuse and guaranteeing accountability. On the transport side, security services have to ensure the availability of functions and elements enabling the transport capabilities. Accountability is highly desirable, although the extent to which privacy concerns are enforced may set some limitations.

Requirements on Mobility: The consumers have already a multitude of devices to communicate through a range of different heterogeneous networks, each one with specific connectivity services and with different mobility approaches. We need thus to inherently support multi-access (i.e., L1/L2 technologies), multi-homing (i.e., several L3 addresses) and multi-domain cases where several business models may exist in parallel. In a such environment, consumers require service continuity or even seamless handover for flows like voice. Likewise, they also need to be always reachable and be provided with consistent and personalised services, i.e. awareness of their location and network capabilities. On the other hand, connectivity services shall be profitable for operators, i.e., Mobility-as-a-service shall be provided only when necessary. In addition, specific procedures such as per-flow mobility anchor selection and activation are needed, depending on a given communication context (type of application, user preferences, terminal capabilities, radio environment, etc.) and on mobility patterns. Finally, the support for decentralised approaches for mobility is also required to cope with the gradual expansion in network capacity.

Requirements on Resource Management: Within the heterogeneity of deployed networks, seamless integration of resources control and management, especially from accesses and edges, becomes also a key requirement. Moreover, to achieve the service-awareness, the context of the application shall be considered as well (e.g., content distribution, cloud computing, real-time communications). Self-organisation and distributed resource management is an important aspect. The abstraction of network resources and features shall enable us to exploit the heterogeneity of technologies on an end-to-end perspective. Virtualisation allows also the flexible sharing and management of resources. The cooperative planning, operation, control and management of connectivity services and technologies shall enable better network efficiency, resilience, scalability and future evolution. It shall leverage advanced features of link technologies, making use of network diversity and aiming at a dynamic and seamless switching between technologies, dependent on flow's requirements. Management of resources shall be dynamic and adaptive to the changes within the networks. The resource management of such wireless networks shall be supported by cognitive radio and spectrum sensing, and mechanisms shall be energy efficient in the management of resources.

3 OConS Architectural Guidelines

Current design paradigms for networking architectures are starting to show their limits because they are lacking the ability to cope with the stressing requirements imposed by nowadays applications and services. The initial design guidelines and principles employed for the Internet endorse (see, e.g., [1]): the connectionless (i.e., best-effort) IP-datagram forwarding, the maximum sharing of the routing information (i.e., routing tables in each router), the end-to-end transport principle where most of the complexity is kept within end-nodes (e.g., TCP, SCTP, HTTP), the modularisation (i.e. layering) with weak cross-layers interactions, the simplicity principle (e.g. cost-effectiveness), and the usage of the IP interface "address" as both locator and name.

Accordingly, some of these principles, which have shaped the current solutions, should be at least revisited in order to see whether they are still capable to deal with the challenges and requirements introduced earlier. In addition, most of the current solutions for managing the connectivity services (e.g., data-transport, routing, mobility, QoS) deal with rather concrete aspects of the whole problem; for example, they are either focusing on the establishment and maintenance of an end-to-end flow (but sometimes still related to specific IP realms), while others are concentrating on the particular issues which affect the core or the access part connectivity.

Thus, in our view, the first architectural design guideline to be followed by the OConS architecture is a holistic approach to the networking. Likewise, the openness, which intrinsically characterizes the OConS approach, also calls for more comprehensive approaches to address the overall connectivity issues from the broadest internetworking perspective.

One of the cornerstones of the OConS approach is to minimize the impacts induced by technology constraints, *aiming at technology independence* to a feasible extent; this spans over both the access part (wireless and fixed) as well as the core network (e.g., switching, routing, interconnection between data-centres, and so on); likewise, the Multi-P (Point/Path/Protocol) paradigm has been coined within OConS so as to reflect this intrinsic characteristic. The OConS architectural framework should thus support the adaptation to the rapid evolution of the communication technologies, its *components need to offer common functionalities*, which can be used independently of the particularities of the underlying technologies or the applications using OConS.

Then, the management of the connectivity services should follow an *autonomous operation*, able to dynamically adapt to various conditions as well as to *cope with various decisions making approaches* (distributed, centralised, mixed, etc.). This automaticity requires, among other things, procedures to discover and negotiate the corresponding services and functionalities.

A straightforward consequence for OConS design was *the choice of a modular architecture*, built following a component-based approach, which can instantiate its different entities according to the particular needs and which can therefore be re-used in difference contexts. By *implementing well-defined interfaces*, this flexible modular design also allows the independent modification and enhancement of each module, while hiding the complexity of the embedded mechanisms (and their evolution) to the users. Likewise, whenever possible, the framework should *ease the reflexive and recursive use of its different methods and services.*

In addition, an *appropriate (including tight) interoperation between layers* is also foreseen within OConS, dynamically coupling the corresponding connectivity services across several layers (e.g. cross-layer mobility management, cross-layer GMPLS instance, etc.); this cross-pollination would also bring in the *context and service awareness*, thus the OConS will be able to tailor its services according to different constraints, e.g., application, energy, cost, QoS, location.

The next paragraphs detail the specific guidelines from different point of views.

Design Guidelines on Openness: the “Openness” motto has various implications and consequences for the design of the OConS framework; this guideline affects all types of connectivity services, and as such (using an well-known example) we should go beyond the current OpenFlow, i.e., not limiting ourselves to the policing/steering of the forwarding mechanisms for a given flow. This also leads to the definition of publicly available interfaces, with standardised functions (primitives), behaviour (sequence of primitives) and formats (encoding of information elements). It finally implies the accessibility to the available connectivity services to any authorised user, making the frontiers between different domains more permeable; however, this has clear impacts on security, e.g., privacy and access control.

Design Guidelines on Routing: one guideline, adopted by OConS, is to split data forwarding (which usually happens on a distributed way) from routing control and policy (mostly a centralised process), with two main facets: (1) both mere data forwarding and routing protocols should be executed in a possible distributed manner; and (2) there should be a clean split between the routing decisions to a (set of) destination(s) when multiple paths are available without involving policies, and the policies themselves. Because the global routeability might not be available for all services and applications, the OConS will also consider the limitations imposed by a given addressing and naming scheme, with implications, e.g., on the size of routing tables, the scalability of routing mechanism, the number of VPN contexts, etc.

Design Guidelines on Transport: as opposed to most of the current models, OConS deals not only with the traditional end-to-end paradigms, but also with hop-to-hop (like in DTNs) or edge-to-edge (e.g., VPLS/OTV/TRILL) approaches. OConS will cover thus several scenarios, like the support of multiple points of attachment, broadcast, multicast, or anycast communications, as well as the connectivity among a set of destination and potential sources (e.g., gathering content from various caches). Likewise, multiple types of congestion control (e.g. window based, rate based, delay based) need to be supported for an application depending on its flows' requirements, as well as different options for reliability on specific paths and/or a specific reordering level. Thus, for the establishment and the management of the connectivity, OConS should not be limited to the control/management of single packets, but also to their different logical aggregation levels, such as: flows, sessions, bearers, paths, etc. However, we are not targeting a connection-oriented approach; we are advocating a connection-emulated approach, enriched with several connectivity services, while still making use of the advantages of the packetised networking (IP, MPLS, and Ethernet).

Design Guidelines on Security: we will follow here general security guidelines, aiming at authentication and authorisation as well as confidentiality, integrity and availability. It is worth saying that connectivity services should be only provided when all involved entities (previously authenticated) have agreed to do so. The implementation of security services shall use suitable cryptography technologies following a security by design, as opposed to security by obscurity. Besides, the goal for selecting an implementation technology shall be to first use existing, well-proven standards, and only develop new solutions if this cannot be avoided. On the other hand, privacy (tightly linked with security) goes beyond traditional requirements, to ensure not only protection of users' data, but to enable the user control of its privacy protection level. We also need to consider the broader case, which targets protection of data belonging to operators, service providers or any entity related to either the use or the provision of OConS services. Hence, the exposed information shall be adapted and filtered to other entities depending on the particular policies, but still assuring the correctness of that information.

Design Guidelines for Mobility: the ultimate goal being to ensure transparent and seamless mobility, the OConS should offer the possibility to instantiate on-the-fly various mobility solutions only when needed; one example of this would be the possibility of establishing on-demand tunnels instead of re-routing. On the other hand, mobility decision entities should be dynamically distributed or chosen, as opposed to the centralised approaches. Mobility support might be confined to a given domain or considered at a global scope, thus leading to various types of resolution/mapping mechanisms (e.g., global, localised, "service-specialised", and so on). On the other hand, mobility services do not have to be restricted to end-terminals, but they could be extended to content-IDs, or processes/virtual-hosts.

Design Guidelines for Resource Management: most of the currently available procedures are based on centralised approaches, i.e., they do not benefit from a closer cooperation among the resource managers. In the OConS we want autonomous resource management mechanisms (that is, able to operate of a self-* way), while supporting a distributed operation, and being able to share the decision processes with other peer-entities. OConS need also to facilitate the interoperation between different entities belonging to different administrative domains (e.g. operators). To achieve a better cooperation/coordination, a modular approach was endorsed within a comprehensive framework, so that the mechanisms can be combined on-the-fly, as they are required for a given networking situation and application/service context. It is worth highlighting that in the OConS framework we assume that the networking/communication resources can be virtualised and thus, we consider them as yet another type of items to be managed.

Design Guidelines for Migration: in the OConS context this guideline deals with the phased introduction and inter-operation of the newer generation subsystems with functionally-comparable subsystems of an older generation. Thus, the OConS need migration paths, describing how to update a given system to the new generation of services and functionalities, without compromising its legacy functionalities. Likewise, in order to allow the newer subsystems to communicate among them throughout the older generation subsystems, it is also desirable that the latter ones implement some extension mechanisms, so as to tolerate the newer features.

4 The OConS Architecture

As we have discussed before, although there has been quite a few number of proposals to cope with the stringent requirements of today's new services and applications, most of them lack of a holistic approach, and are tied to rather specific scenarios or technologies. Opposed to that, the SAIL project is fostering a flexible and scalable approach, starting from the requirements stated in Section 2 and following the principles which were discussed in Section 3.

One of the main difficulties which needs to be coped with is the wide range of technologies which are involved in any communication, from the access part to the core networking techniques. Hence, finding a common denominator which might be used so as to properly describe all the involved procedures is the first thing to be identified, leading to the non-tight framework which is deemed necessary. In this sense, from a high level perspective, we can say that most of the processes which are envisaged to be part of the OConS architecture can be categorized into three main phases:

1. Collecting information.
2. Taking decisions on the basis of such information.
3. Enforcing such decisions, by instantiating the appropriate modules.

This stepwise vision of connectivity procedures, which is depicted in Figure 1, must be also reflexive, since the enforcement of a decision could also trigger the complete cycle again. Using these three phases as guidelines, the OConS architecture has been conceived to ensure the flexibility which is required to integrate mechanisms and procedures having great differences between them. In particular, there are two main groups of functionalities: those which deal with the management of the connectivity, and those which are more in charge of the enforcement of particular connectivity services.

4.1 The OConS Functional Elements

In order to mimic the vision which was presented before, the OConS embraces three functional elements, which aim at being independent of and abstracted from any layer of protocol which might be involved in the communication procedure. These elements, which are briefly introduced below, are assigned to real nodes and entities within the network on a dynamic and flexible way. This flexible and open approach allows the support of a large number of different configurations, topologies and scenarios. The three entities can be instanced on one single entity or be distributed between several nodes, fostering a distributed operation.

1. **Information management Element (IE).** These elements are spread within the whole network (end-user devices, access elements, network nodes, etc.) and collect any relevant piece of information, which is afterwards delivered to the interested entities. As was also proposed in [7], the information might be preprocessed before being delivered; furthermore, OConS shall support different subscription and request strategies. As will be discussed later, the information is not restricted to elements which can be collected within the network (QoS, QoE, etc.), but could also include policies and preferences from the users, operators, services, etc.
2. **Decision making Element (DE).** It uses the information gathered by the IEs and takes decisions accordingly. The decision might be constrained to a single entity (e.g. a handover decided by a node within the network) or be taken by a distributed decision mechanism.
3. **Executing and enforcement Element (EE).** In most of the cases, a decision taken by the DE leads to some action which might be executed and enforced by some entities within the network; therefore, OConS shall also include the means to handle this enforcement, which usually would not be restricted to the node which originally took the decision.

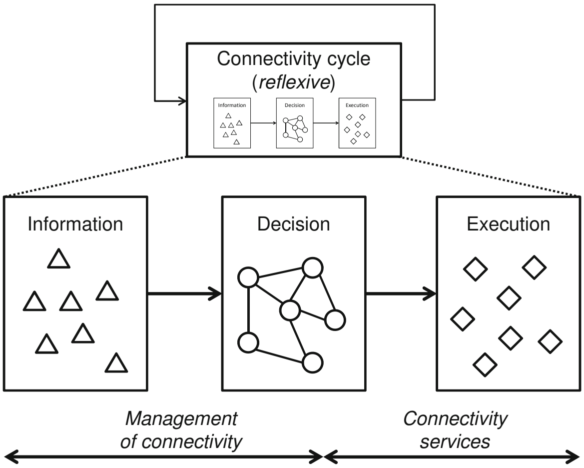

Fig. 1. The three phases in the connectivity cycle

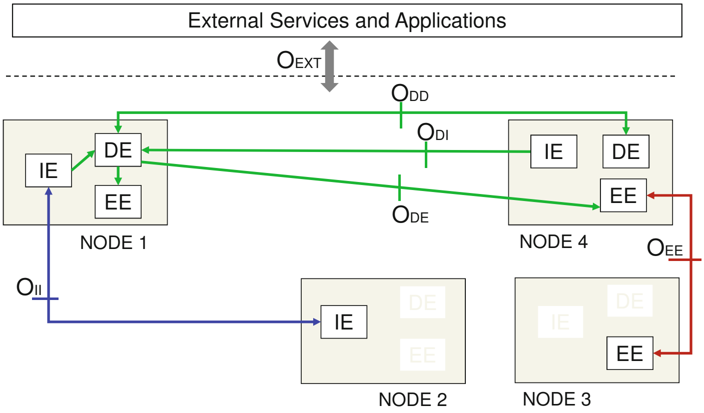

Fig. 2. The OConS architecture and interfaces

Figure 2 depicts the OConS architecture, identifying the interfaces which have been established between the various functional elements, and to/from external services and applications; it is worth saying that we have not made any real difference in the specification of peer internal and external interfaces (they might differ in their implementation details, though). The represented nodes are only a sub-set of possible node types, which may consists of one to many instances of each of three functional elements, IE, DE, and EE. As illustrated, a node may retrieve information, take decisions and execute those decisions (NODES 1 and 4), or may only be a monitoring (NODE 2) or an execution node (NODE 3). However, any combination of the functional elements is allowed. In Section 4.4 an example of mapping a generic mechanism into the architecture is presented. In the next subsection, the definitions for the interfaces are introduced.

4.2 OConS Interfaces and APIs

The OConS internal (between the aforementioned functional entities) interfaces assume a clear distinction between the control and data planes. In this sense, most of the functionalities which are envisaged to be offered by the OConS framework are mostly based on control and management operations, thus leading to a broad range of different messages and functionalities. In this sense, some of the foreseen operations are common to all the interfaces (request/response exchange, discovery procedures, etc.).

On the other hand, when it comes to the data plane, OConS restricts to the transfer of data between peer EEs, and no other interface is deemed necessary.

We also provide some initial set of messages to be exchanged with other functional entities, which might be willing to make use of the services provided by OConS; special attention is paid to the interoperation with the other two pillars of the SAIL project, namely the Network of Information (*NetInf*) and Cloud Networking (*CloNe*).

Internal Interfaces

In order to specify these interfaces, we assume that all OConS entities have names, which can be resolved into the appropriate addresses and locators; furthermore, it is also assumed that available technologies are able to handle the required bootstrapping process and that a communication can be always established between two, or more, OConS entities.

Regarding control and management functionalities, it is important to highlight that all the interfaces share the discovery functionality, which requires the exchange of a set of common messages so as to locate OConS entities and to find out which are their capabilities. Besides, the interface between the DE and the IE comprise messages to configure the operation of the IE (for instance, to subscribe to particular events or situations), to request information, or to send notifications (upon certain pre-configured situations). In order to enable distributed decision processes, an interface is needed between peer DEs; as DE needs to send execution and enforcement commands, an interface towards the EE is also required. Finally, an interface will be used between peer IEs, so as to enable distributed collection of information (this might be used, for instance, in spectrum sensing techniques), and another one would be specified between EEs, so as to manage some particular actions which shall lie under their responsibility. Note that the interface between IE and EE is not deemed necessary.

On the other hand, the interfaces for data transmission are only limited to the transmission/reception of actual data between EEs, which is the only OConS entity which handles the data of the applications/services.

External Interfaces

OConS has the main goal of easing the process of establishing communication paths for application and service flows. In that sense, it can be seen as as improvement of the traditional BSD socket interface, so that an application (i.e., OConS-aware) could benefit of the functionalities offered by OConS so as to send and receive data. Besides, a number of control and management messages are also foreseen, mostly related to the registration/deregestration procedures, as well as to the establishment of paths as a means to send/receive data flows using the appropriate connectivity services.

4.3 Information and Data Model

As has been already mentioned, a cornerstone of the OConS operation is the decision-taking procedure, which is based on the pieces of information gathered by the IEs. Therefore, it is essential to define a proper data model, able to adapt to the wide range of the information elements which are foreseen to be exchanged between IEs and DEs. This information is structured as follows.

- **Resources.** The resources can refer to network (links, nodes, etc.) or end-user resources (the latter embracing terminal and devices), might be dynamic (varying over time). Each of the resource items is characterized by a set of attributes, which depend on the type of resource.

- **Context.** Context to those constraints which are established by the particular situation in which the connectivity needs to take place (scenario, location, mobility, etc.).
- **Requirements.** They can be from either the application or the user, and they usually refer to the type of service they would expect (price, QoE, etc.).
- **Policies and preferences.** They can refer to the service, the user or the operator, and they normally describe static rules which should be followed when establishing the connectivity.

4.4 Example of Application

As an example, the mapping of the OConS architecture is presented for a use case dealing with creating and sustaining the connectivity in wireless challenged networks. Consider several heterogeneous wireless nodes willing to build a multi-hop network in order to provide the end-users with the connectivity between them and towards a fixed Internet infrastructure. This communication environment is often under adverse conditions, e.g., expectations of connectivity between certain nodes no longer holds, or congestion is experienced on some links because of the multiple simultaneous requests from the crowd. The sharing and optimization of nodes' resources in a cooperative and self-organized way enables the distributed management of the whole network:

- Newly added nodes self-configure themselves in a plug-and-play fashion.
- Nodes regularly self-optimize their resources in response to network changes.
- In the event of a node failure, self-healing mechanisms are triggered in the surrounding nodes to alleviate gaps of connectivity, coverage or capacity.

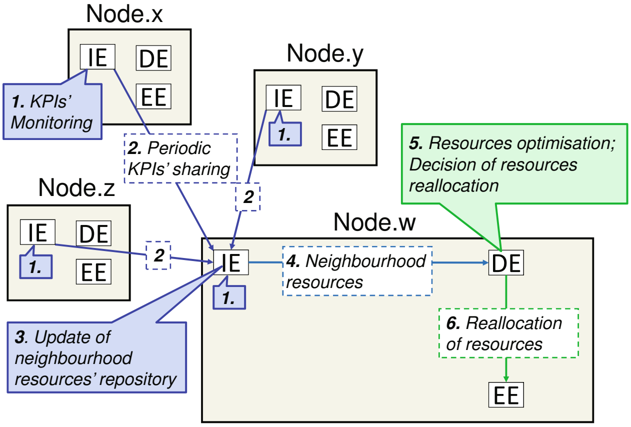

Fig. 3. Mapping of the OConS architecture on a self-organized challenged network use-case

The different steps of the self-optimization mechanism, mapped into the OConS architecture (i.e., functional entities and their interfaces) as depicted in Figure 3, are further described below:

- Nodes monitor a set of Key Performance Indicators (KPIs), such as channel, load, data-rate, SINR and power (step 1 of Figure 3).
- KPIs are shared within the node's neighborhood, through message broadcast IE-IE interface (step 2).
- Neighborhood KPI's information is collected and compiled by each node, being dynamically updated (step 3).
- Based on the collected information (step 4) and on specific strategies, a node takes a decision (IE-DE) (step 5).
- The decision is then enforced locally within a node (DE-EE), or communicated remotely to another node for the enforcement (DE-EE) (step 6).

5 Conclusions and Outlook of Future Work

In this paper we have presented the OConS architecture, developed in the framework of the *SAIL* project, as a means to overcome some of the limitations imposed by legacy connectivity solutions, yet maintaining realistic migration paths. OConS aims at providing the appropriate set of functional entities and their interfaces, so that any technique, protocol, or algorithm, can be adapted to fit into its framework. Furthermore, it does not focus on a specific area, but it fosters a holistic approach, paying attention to both access (e.g., mobility) and core networking issues (e.g., data-centre interconnection); in this sense, it goes beyond other initiatives, which have looked at more specific problems.

In order to achieve such degree of flexibility, we have identified a common denominator for most of connectivity operations, being a reflexive cycle which embraces: (1) the gathering of information; (2) the decision making on the basis of such information; (3) the enforcement of such decisions to the appropriate network elements. OConS allows this cycle to be executed both reflexively and recursively and it fosters the cooperation between peer entities for any of the corresponding connectivity services (for instance, to implement a distributed decision mechanism).

As an illustrative example, we have shown how a particular problem can be addressed by means of the proposed framework; thus, we have applied the OConS approach to address the resource management in Wireless Mesh Networks, showing that it can effectively adapt to various types of challenges.

This architectural framework sets the basis for various lines of future work. First we will analyze the performance of various techniques, algorithms and protocols, which might benefit from the functionalities which are brought about by the OConS framework; in particular, special attention will be paid to the interrelation with the two other SAIL pillars: the Network of Information and Cloud Networking, and how they could take advantage of the OConS services. Furthermore, simulation and prototyping activities will be also pursued, with the main goal of assessing the feasibility of the proposed architecture and comparing its mechanisms with the legacy networking solutions, thus providing a sound migration strategy at the end.

Acknowledgements. The authors would like to express their gratitude to the European Commission for its funding through the "Scalable and Adaptive Internet Solutions", SAIL Project (FP7-ICT-2009-5-257448).

References

1. EC FIArch Group: Fundamental limitations of current internet and the path to future internet (March 2011), `http://ec.europa.eu/information_society/activities/foi/docs/current_internet_limitations_v9.pdf`
2. Farrel, A., Vasseur, J.P., Ash, J.: A Path Computation Element (PCE)-Based Architecture. RFC 4655, Informational (August 2006), `http://www.ietf.org/rfc/rfc4655.txt`
3. Ford, B., Iyengar, J.: Breaking up the transport logjam. In: Proceedings of ACM HotNets (2008)
4. Guillemin, F., et al.: Architecture of a generic path. Deliverable D5.1, EU-FP7 4WARD project (January 2009)
5. Han, H., Shakkottai, S., Hollot, C.V., Srikant, R., Towsley, D.: Multi-path tcp: a joint congestion control and routing scheme to exploit path diversity in the internet. IEEE/ACM Trans. Netw. 14, 1260–1271 (2006)
6. IEEE: Standard for local and metropolitan area networks-part 21: Media independent handover (2009)
7. Makela, J., Pentikousis, K.: Trigger management mechanisms. In: 2nd International Symposium on Wireless Pervasive Computing, ISWPC 2007 (February 2007)
8. Mannie, E.: Generalized Multi-Protocol Label Switching (GMPLS) Architecture. RFC 3945 (2004), `http://www.ietf.org/rfc/rfc3945.txt`
9. McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., Turner, J.: Openflow: enabling innovation in campus networks. SIGCOMM Comput. Commun. Rev. 38, 69–74 (2008)
10. Niebert, N., Schieder, A., Zander, J., Hancock, R.: Ambient Networks: Co-operative Mobile Networking for the Wireless World. Wiley (2007)
11. Pentikousis, K., Agüero, R., Gebert, J., Galache, J.A., Blume, O., Pääkkönen, P.: The Ambient Networks heterogeneous access selection architecture. In: Proceedings of the 1st Ambient Networks Workshop on Mobility, M2NM (October 2007)
12. Perlman, R.: Rbridges: transparent routing. In: 23rd Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 2004, vol. 2 (March 2004)
13. Radriamasy, S., et al.: Mechanisms for generic paths. Deliverable D5.2, EU-FP7 4WARD project (May 2010)
14. Suciu, L., et al.: Architectural concepts of connectivity services. Deliverable D4.1, EU-FP7 SAIL project (July 2011)
15. Woesner, H., et al.: Evaluation of generic path architecture and mechanisms. Deliverable D5.3, EU-FP7 4WARD project (June 2010)