Generic Connectivity Architecture for Mobility and Multipath Flow Management in the Future Internet

Amanpreet Singh[1], Christoph Nass[1], Andreas Timm-Giel[2], Peter Schefczik[3],
Horst Roessler[3], and Michael Scharf[3]

[1] Center for Computing and Communication Technologies (TZI),
University of Bremen, Germany
[2] Institute of Communication Networks, Hamburg University of Technology, Germany
[3] Bell Labs Germany, Alcatel-Lucent, Germany
`{aps,chn}@comnets.uni-bremen.de, timm-giel@tuhh.de,`
`{peter.schefczik,horst.roessler,`
`michael.scharf}@alcatel-lucent.com`

Abstract. With the evolution of the Internet, the vast majority of the traffic is generated by information-centric applications, which would benefit from enhanced data transport paradigms. This paper presents the development and implementation of the Generic Connectivity architecture, a new communication flow abstraction that is based on the Generic Path architecture developed within the European Research Project 4WARD. The Generic Connectivity mechanisms allow for a high degree of flexibility by covering both existing and new protocol paradigms, which are particularly beneficial for wireless access networks. This paper shows that the Generic Connectivity architecture can realize new network mechanisms beyond the features of the current Internet protocol architecture. It is thus a promising clean-slate approach for the Future Internet. The relevant aspects of the architecture are implemented with the OMNET++ 4.0 network simulation tool. Using simulations, the advantages of the Generic Connectivity architecture are shown for several new use cases, including an adaptive protocol selection, mobility, multicast and multipath connectivity over heterogeneous wireless networks. Furthermore, it is also demonstrated that the architecture inherently supports guaranteed Quality-of-Service (QoS) agreements and traffic distribution over dynamic channels.

Keywords: Future Internet, Mobility, Multipath, Quality of Service.

1 Introduction

The current Internet architecture is challenged by the rapid growth of the number of nodes and the increasing traffic volume, as well as the fact that more and more content is accessed via mobile devices. These are major issues for the evolution towards the Future Internet [1].

The 4WARD project [2] addresses these challenges by a "clean-slate" architectural approach. 4WARD provides a flexible framework that allows a number of different networks to co-exist and inter-operate by realizing multiple virtual networks, e.g.

K. Pentikousis et al. (Eds.): MONAMI 2010, LNICST 68, pp. 1–13, 2011.

information-centric networks. Moreover “default-on” management capabilities within the network are incorporated and the network path is made an active element called the "Generic Path".

A Generic Connection is a new networking communication abstraction. It is set up between two or more communicating end-points and organizes the cooperation between nodes for a wide range of communication services. Unlike the existing Internet architecture, the Generic Connection inherently integrates mobility, multipath transport, multicast support, as well as QoS mechanisms.

This paper shows the benefits of the Generic Connectivity architecture and investigates how it could be implemented in a Future Internet, using the 4WARD concepts as a basis. As a proof-of-concept, a discrete event simulator implementation of the newly developed architecture is presented. The key advantages of the new architecture are demonstrated by experiments with adaptive error and flow control and multipath traffic distribution over heterogeneous wireless access networks.

The rest of this paper is structured as follows: Related ongoing work in Future Internet research is briefly summarized in section 2. The Generic Connectivity architecture is presented in section 3. The flexibility of the Generic Connectivity architecture can be seen by applying concepts of mobility, multipath and multicast in section 4. The simulation tool implementation and evaluations of different scenarios are depicted in section 5 and 6, respectively. Finally, section 7 concludes the paper and provides an outlook.

2 State of Art

There are numerous ongoing research activities for designing the Future Internet, e.g., the NSF Future Internet Design (FIND) program and the Global Environment for Network Innovations (GENI) platform in the US. The latter’s purpose will be to implement and test a wide range of research proposals in distributed global testbeds.

Sensor and mobile wireless networks are a key challenge for Future Internet design. This has also triggered research activities on fundamentally new protocol architectures, for instance in the European 4WARD project. The 4WARD approach to mobility is described in [3].

A clean slate approach combining routing with content data was triggered by Van Jacobsen and others [4]. The content centric network (CCN) proposal makes mobility management for certain services easier by putting content Ids into the forwarding table of the routers and making the content itself move. A related approach is targeted by network information objects within the 4WARD project.

Both trends result in a need for more flexible data transport mechanisms. This requirement has also been identified in [5], and is addressed there by introducing a separate flow layer and factoring endpoint addressing into a separate endpoint layer.

3 Generic Connectivity Architecture

A generic architecture for communication paths in the Future Internet must support the growing diversity of applications and network technologies, allow multiple points of attachment and maintain seamless connectivity for mobile hosts and networks. Due to the varied requirements, it is not possible to envision a single transport solution but a family of communication paradigms differing in their characteristics and types.

The Generic Path (GP) framework developed in the 4WARD project abstracts and generalizes a number of transport connections from physical wires, wireless mediums, optical fibers and virtual connections. In addition to the data transfer capabilities, the GP architecture inherently allows for data transformation such as aggregation, encapsulation, encryption, translation, coding and transcoding [6].

3.1 Overview and Terminology

The Generic Connectivity (GC) architecture extends the object-oriented programming technique of inheritance (consisting of base classes, methods and procedures) proposed in the 4WARD project. Similar to the GP architecture, the GC architecture also enables modular GC services, allowing for a recursive architecture where complex or advanced (higher level) GC services can be obtained from simple and basic GC services. The different objects of the Generic Connectivity architecture are shown in Fig. 1 and are explained in the following:

Information Object (InfObj) - InfObj is a client of a Generic Connection, capable of consuming/producing information.

Entity (ENT) - An Entity may be a process, a thread in a process or a set of processes. It can communicate with other Entities of the same compartment by means of a Generic Connection.

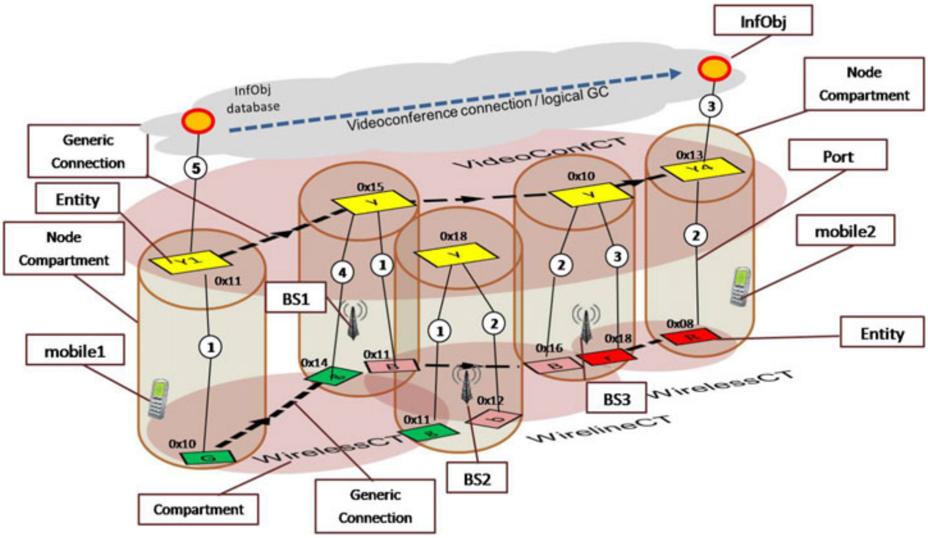


Fig. 1. Simplistic Generic Connection Scenario

Compartment (CT) - A compartment has its own set of protocols for operations such as routing, forwarding, authentication, security, authorization, etc. and may even follow strict policy enforcements. In addition, addressing is unique within the scope of a compartment and therefore a compartment can be seen as a ‘name space’. Like the Generic Connection, different types of Compartments are described by inheritance in the object-oriented model.

Node Compartment (NodeCT) - A node compartment is composed of the software capable of supporting tasks that can all atomically reference the same memory space [6]. A node compartment manages a name space to address Entities. A Node Compartment can relate to a processing system as defined in the Network Inter Process Communication Architecture (NIPCA) of [7].

Ports - A Port is a handle to an Entity, an identifier local to the Node Compartment.

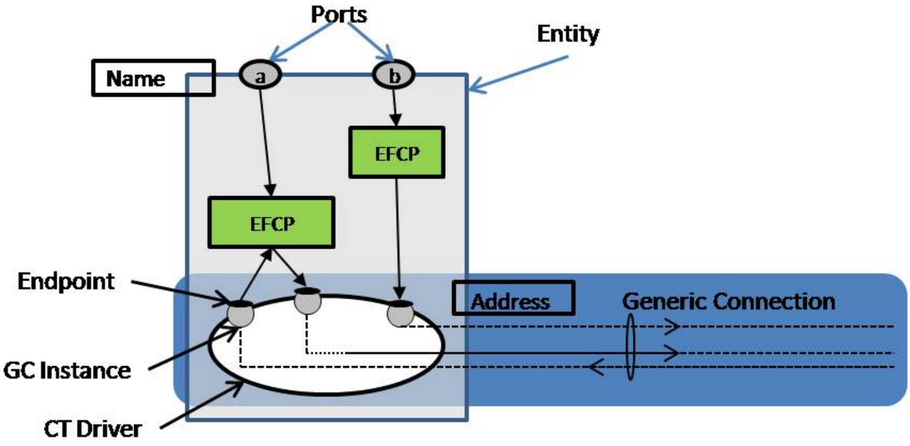


Fig. 2. The Entity Internal Structure

While the GP architecture [6] does not describe node internal structures, the following additional structures were defined within the GC architecture for a real implementation (refer to Fig. 2):

Generic Connection Instance (GC Instance) - Data transfer including forwarding and manipulation of data is executed by GC Instances. A GC Instance keeps the local state information of a GC. It is a thread or process executing a data transfer protocol machine. GC Instances are created by an Entity and may access shared information of that Entity.

Error and Flow Control Protocol (EFCP) - EFCP takes care of the reliable transmission of the messages over the Generic Connection.

Endpoint (EP) - A Generic Connection terminates at an Endpoint within an Entity. The EFCP injects or retrieves packets from the GC via the Endpoint.

Compartment Driver (CT Driver) - CT Driver identifies the Entity within a compartment. An Entity can only join a single Compartment at a time.

3.2 Naming and Addressing

In the Generic Connectivity architecture framework, the entities are assigned names and compartment specific addresses. To resolve an entity, the name resolution is done. The entity always keeps its name even if it moves and joins a new compartment, in which case the same entity will have a new address. Within a Node Compartment also the different entities have to be identified by addresses or some other forms of identifiers. In addition, there is also a need to uniquely identify a port with port numbers.

3.3 Generic Connection Setup

To account for the design goals along with the optimal combinations of the communication protocols, GC services are compartmentalized [6]. Therefore, when an entity wants to set up a Generic Connection in order to initiate communication, it needs to be a member of the appropriate compartment. In order to do so, the entity gets the compartment information from the Node Compartment, searches for the specific compartment or creates the compartment itself and advertises it. In addition to being a namespace, the compartment can be seen as a signaling control plane which provides a topological view and specifies the rules to be followed by its members. Once the entity is a member of the compartment, a Generic Connection can be established as the compartment can obtain the resource information from the underlying and/or neighboring compartments.

4 Mobility, Multipath and Multicast in the Generic Connectivity Architecture

Mobility can be classified into different types like device mobility, network mobility, session mobility, etc. The Generic Connectivity framework can be extended by specialized mobility management mechanisms with respect to the specific characteristics of a compartment. In the following, the mobility, multipath and multicast concepts of the Generic Connectivity architecture are summarized.

The *mobility* solution is shown in Fig. 3. Therein, a Provider compartment is distributed over wireless, access and IP compartments. The mobile device, assumed to be capable on supporting parallel wireless connections, is first a member of the wirelessCT1 and as it moves it appears in the vicinity of another compartment wirelessCT2 it will have to perform a handover from wirelessCT1 to wirelessCT2. Since wirelessCT2 is a new compartment, the entity in the Provider compartment instantiates a new entity for the wirelessCT2 to set up a Generic Connection in the wirelessCT2. The Generic Connection in the Provider compartment is unaware as it will not identify anything whereas the Provider compartment is aware that it now has more resources and more options. In order to use the 2nd available path, a new Generic Connection needs to be established within the access and IP compartments. This *mobility* solution of utilizing multiple paths is of the form of “make-before-break” mobility management.

For uplink traffic, the handover is realized in the GC architecture by switching ports to the wireless compartments. Different from today’s networks, this switching uses the same mechanisms for a handover between two wireless compartments with the same technology or for an inter-technology handover. For downlink traffic, the optimal point from where the downlink traffic is diverted to the new entity needs to be identified.

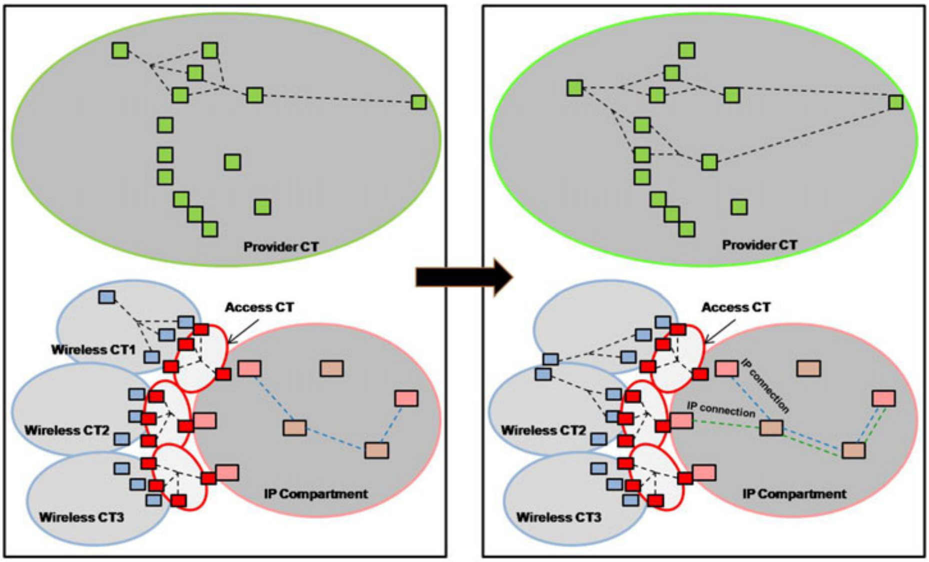


Fig. 3. Mobility – Compartment View

One of the novel features provided by the Generic Connectivity framework is the *multipath* transport. The Generic Connection is spread over multiple links to form the end-to-end transport connection within a compartment. While establishing the Generic Connection, it may request multipath transport from the compartment, if available. In doing so, the Generic Connection can choose amongst the various combinations that exist for multiple paths.

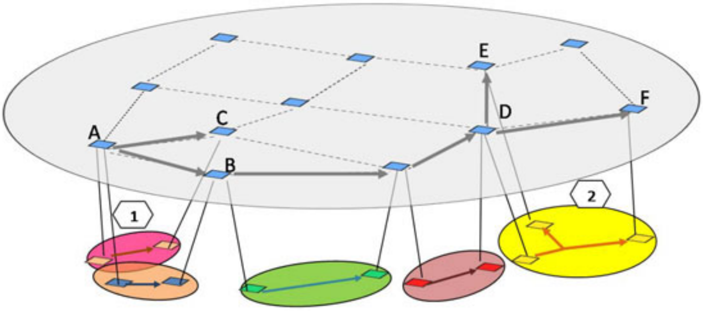


Fig. 4. Multicast – Compartment View

In Fig. 4, an example of a *multicast* Generic Connection is illustrated. The interesting aspect is that the Generic Connection in the higher layer compartment can span over multiple lower layer compartment types. On the left hand side, the realization of multicast requires that the entity *A* has to duplicate the packet and send them out on both ports. In contrast, on the right hand side, the underlying compartment is a broadcast compartment or already contains a multicast Generic Connection. Hence, the entity *D* only has to send the packet to a single port and the lower compartment will forward it to both entity *E* and *F*.

5 Simulation Tool Implementation

In order to show the feasibility of the developed Generic Connectivity concept, a demonstrator based on the network simulation tool OMNeT++ 4.0 [7] was developed. It implements the GC concept and methods for testing and validating the characteristics of the Generic Connectivity architecture.

5.1 Simulated Scenario

Fig. 5 depicts the network topology that was simulated, including a video conferencing scenario. It is assumed that the mobile1 is within the range of both base stations bs1 and bs3. The connectivity between the mobile1 and the two base stations bs1 and bs3 is represented by link-A and link-B, respectively.

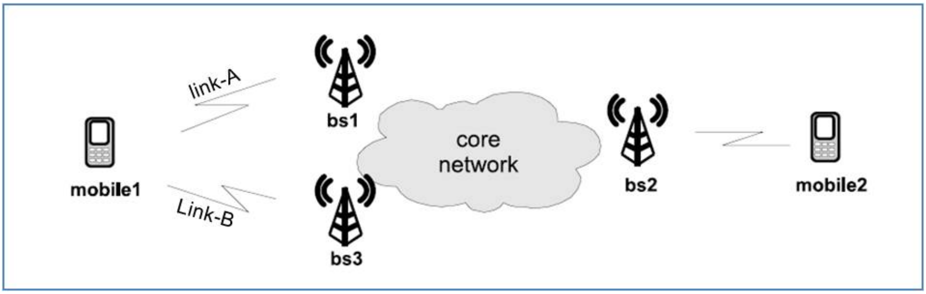


Fig. 5. Simulated Network Topology

5.2 QoS Constrained Multipath Approach

To demonstrate the flexibility of the Generic Connectivity architecture, a heuristic multipath approach for two paths is implemented, where a minimum bandwidth usage policy is used to distribute the traffic with respect to the link quality (packet loss probability) to achieve a pre-defined QoS constraint (packet loss ratio, PLR). To achieve the target PLR, a simple mechanism of duplicating traffic is used. The minimum bandwidth will be used if all traffic can be sent without any duplication. If the link quality is not good enough data duplication is performed. This heuristic algorithm always exploits the best quality path (with least packet losses) to the maximum. If the traffic exceeds the available bandwidth on the best path, the remaining traffic is sent over the second best path.

5.3 Adaptive Error and Flow Control Selection

In the current Internet transport mechanisms, the Error and Flow Control Protocol (EFCP) is integrated into the Transmission Control Protocol (TCP) and cannot be changed with the changing network dynamics. Due to the modular design, the Generic Connectivity framework is much more flexible. This flexibility allows having an adaptive EFCP mechanism within a Generic Connection entity. For the simulations, a *Simple* EFCP module and a *Stop-and-Wait (SnW)* EFCP module were implemented. In Fig. 6, the wireless entities at the mobile1 and base station bs1 (internal-view) are presented. In order to be able to adaptively change the EFCP model, the management module in the entity has to monitor the performance on the link and take appropriate actions depending on the decision algorithm. Once the mobile1 decides to switch between the EFCPs, eventually it has to inform the receiving base station bs1 using a management Generic Connection.

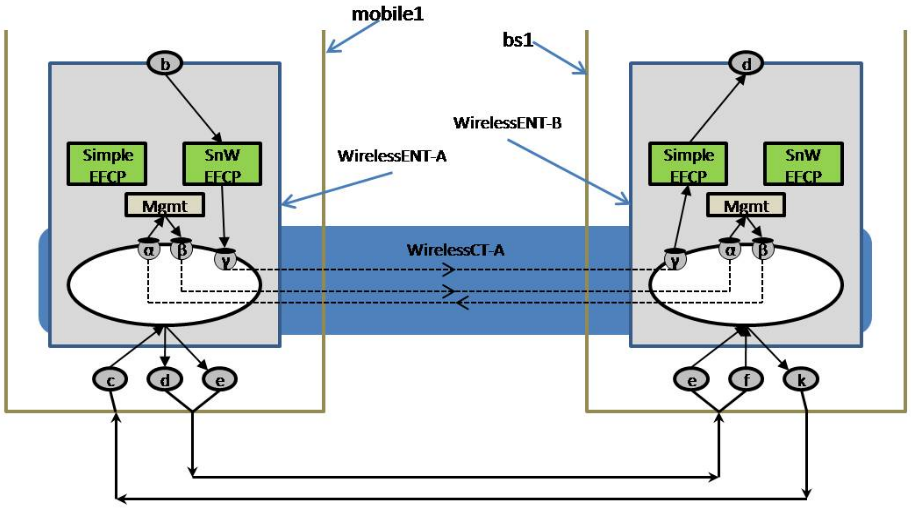


Fig. 6. Wireless Entity view for Adaptive EFCP

5.4 Dynamic Flow Management

For the video conference application considered in the simulated scenario there are two streams – audio and video, as shown in Fig. 7. Within the Generic Connectivity framework, these two data streams can be handled separately by having multiple Generic Connections established for them, even though they are part of the same application data stream. This is another feature that cannot be easily realized in the current Internet.

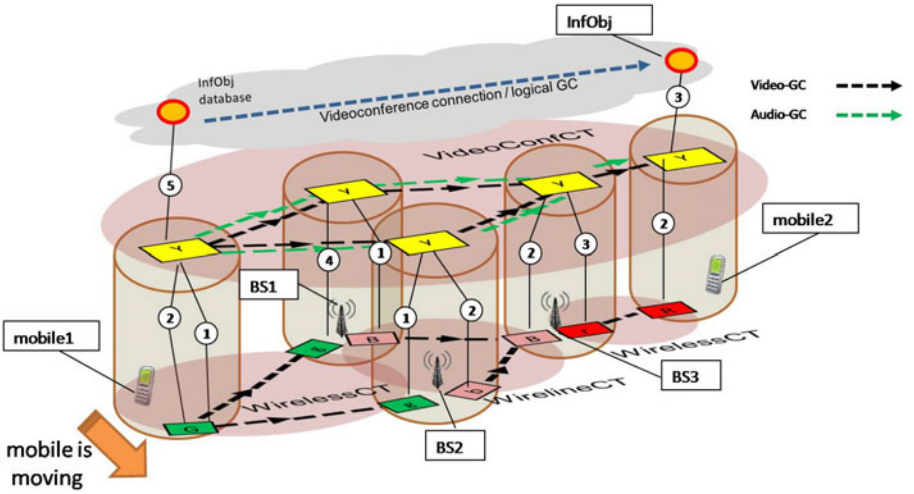


Fig. 7. Audio & Video Multipath Generic Connectivity Management

6 Simulation Results

The following sections present example results for the novel features of the GC architectures and its benefits compared to the existing Internet protocol stack.

6.1 Adaptive Error Correction Mechanisms

For the adaptive EFCP simulations, a packet loss ratio threshold of 0.05 was selected. When the calculated packet loss ratio (depicted in Fig. 8) was less than the set threshold, *Simple* EFCP was used, otherwise *Stop-and-Wait* EFCP was used. The overall packet loss ratio obtained for the adaptive EFCP is presented in Fig. 9and it can be seen that the performance is consistent over the entire simulation period even though the channel quality on link-A was dynamically changing (Fig. 8). In contrast, TCP would always retransmit data, even if this is not required.

6.2 Flexible Multipath Flow Management

In the following, the GC multipath features are illustrated. Fig. 10 depicts the assumed variable packet loss probability of the two links link-A and link-B as mobile1 moves in the left hand side wireless compartment (Fig. 7).

The audio traffic has a data rate of 100 kbit/s and the packet size is 1 kbyte. The target packet loss ratio to be achieved for the audio traffic is set to be 0.025. On the other hand, the video traffic data rate is 1 Mbit/s with the packet size being 2 kbyte and a target packet loss ratio of 0.04. The resources allocated to the audio Generic Connection are 150kbit/s on link-A and also 150kbit/s on link-B. The rest of the resources are allocated to the video Generic Connection i.e., 1.05Mbit/s on link-A and 1.25Mbit/s on link-B.

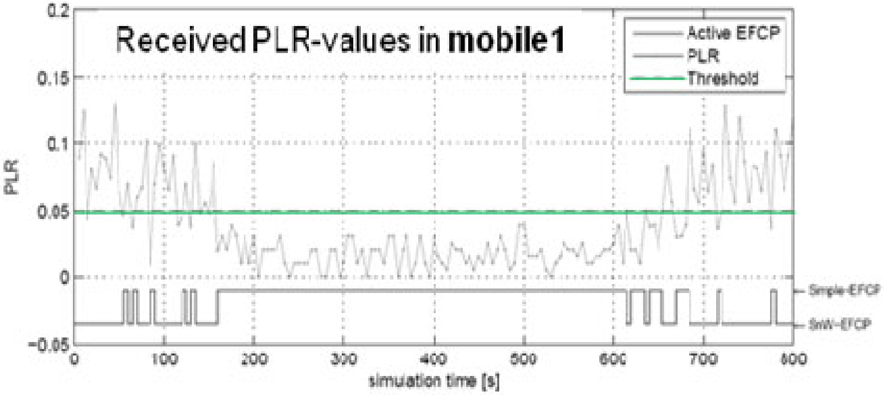


Fig. 8. Adaptive EFCP Switching

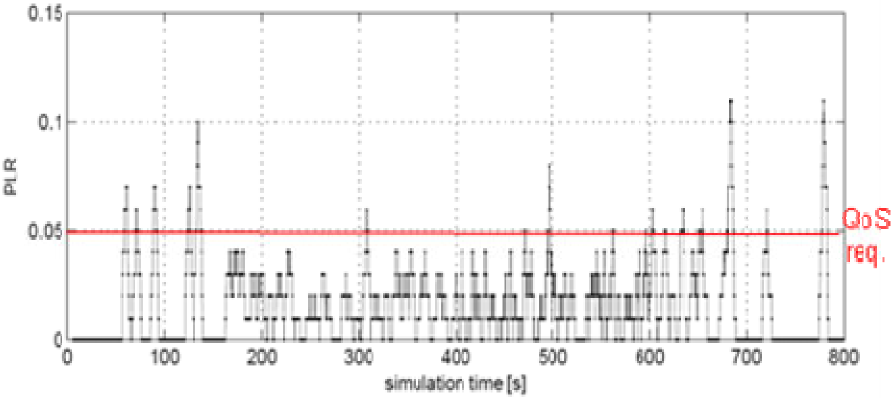


Fig. 9. Packet Loss Ratio seen by the Generic Connection

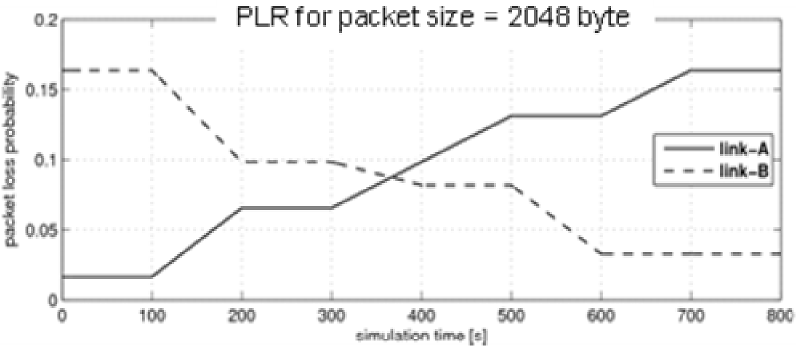
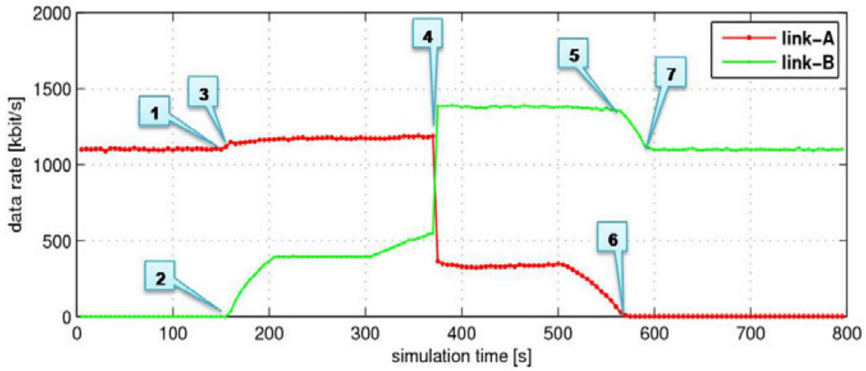


Fig. 10. Packet loss probability in Left-Hand Side Wireless Compartment



1. Video-GP begins to send duplicated packets on link-A
2. Video-GP begins to send duplicated packets also on link-B
3. Audio-GP begins to send duplicated packets on link-A.
4. Link-B is now the "better link"
5. Audio-GP sends single traffic only
6. Video-GP only uses better link
7. Video-GP sends single traffic only

Fig. 11. Combined Data Traffic over the Left-Hand Side Wireless Compartment

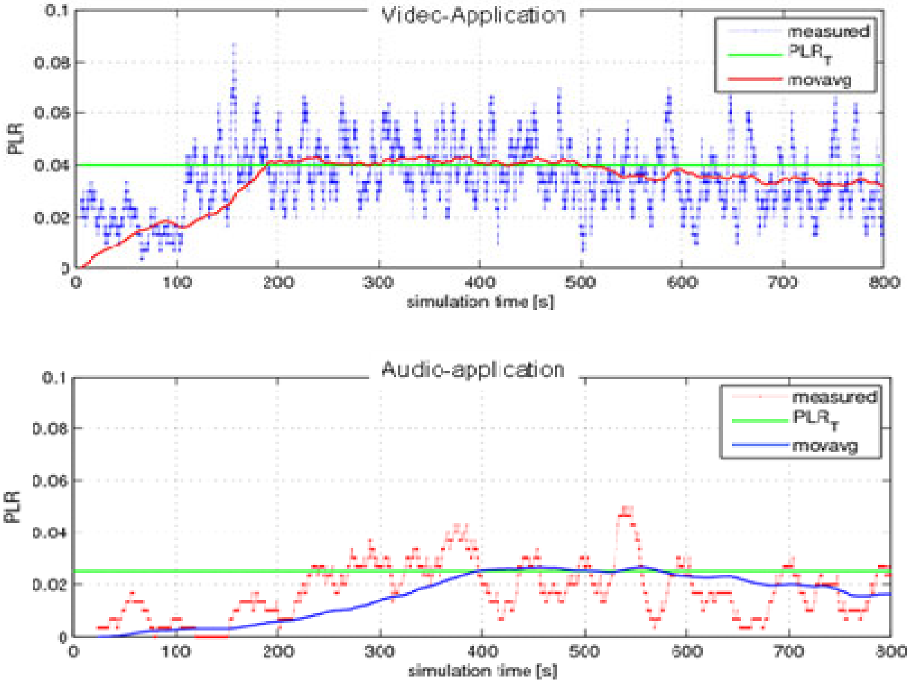


Fig. 12. Loss Ratio for Generic Connections

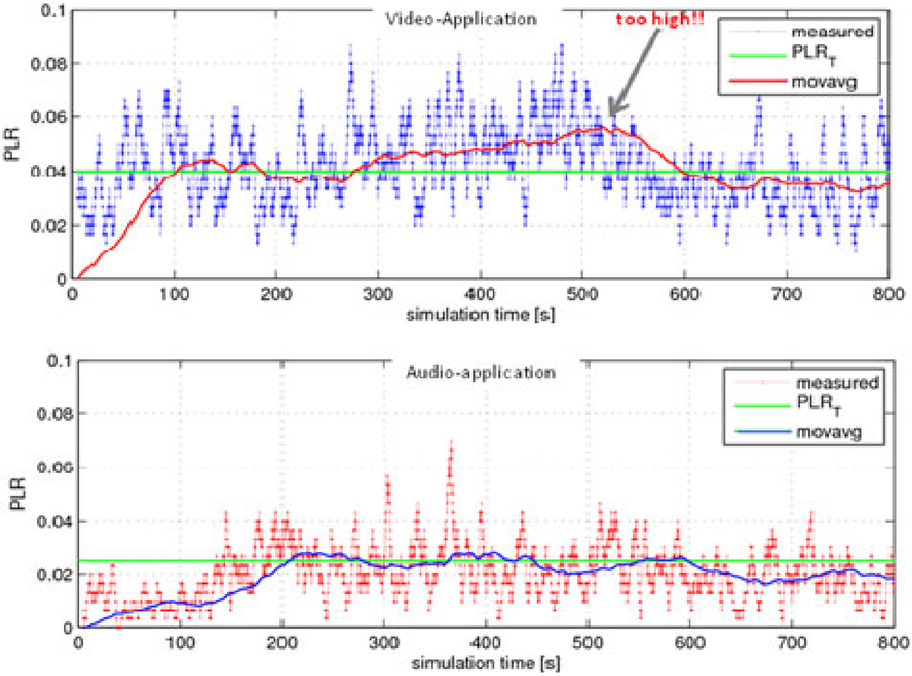


Fig. 13. Packet Loss Ratio for Generic Connections in Bandwidth Limited Scenario

As depicted in Fig. 11, initially both the video and audio Generic Connections use the same link-A for all their traffic. The video traffic requires duplication due to its target PLR and decreasing link-A quality (earlier than the audio traffic) therefore, it switches to duplication mode first and then very quickly starts using additional the resources on link-B (after link-A is fully loaded). After approximately 370s, link-B is the better link and hence it is utilized to the maximum by the two Generic Connections and small portion of data traffic is put on link-A. As the link-B's quality improves the duplicated traffic on link-A is reduced and finally only link-B is used alone by both the Generic Connections.

Fig. 12 depicts the obtained packet loss ratio for both the audio and video Generic Connection and it can be seen that the computed moving average is always in line with the target packet loss ratio (PLR_T).

Finally, a bandwidth constrained example is presented. Now the audio Generic Connection has a higher priority than the video Generic Connection during the resource allocation of the GC architecture. The audio traffic data rate is now 500kbit/s and the allocated bandwidth on both link-A and link-B is 600kbit/s. The video traffic data rate is still 1Mbit/s while the allocated bandwidth on link-A and link-B is only 600Kbit/s and 800Kbit/s, respectively. The packet size and the target packet loss ratio are same for the two Generic Connections as in case of the previous example.

Fig. 13 depicts the packet loss ratio seen by the two Generic Connections. As the video Generic Connection lacks resources, the packet loss ratio is higher than the target value during the simulation run. In contrast, for the audio Generic Connection, there was enough available bandwidth and hence it is always able to conform to its target packet loss ratio. Such a traffic differentiation is still hardly possible in the Internet.

7 Conclusion and Outlook

In this paper, the Generic Connectivity concept was briefly introduced and it was shown that this clean-slate protocol framework is very flexible and powerful. The feasibility and advantages of this framework are demonstrated in several scenarios. The results show that the Generic Connectivity mechanisms support innovative networking paradigms that cannot easily be realized by the current Internet protocol architecture, such as the automatic cross-layer adaptation of error correction mechanisms or flexible per-flow multipath routing over heterogeneous access networks.

Further work is needed to address some remaining research issues such as the design of the signaling mechanisms to realize the Generic Connectivity, as well as further extensions of the supported protocol mechanisms, e. g., for resource management or incremental deployment.

References

1. Banniza, T.-R., Boettle, D., Klotsche, R., Schefczik, P., Soellner, M., Wuenstel, K.: A European Approach to a Clean Slate Design for the Future Internet. Bell Labs Technical Journal (2009)
2. 4WARD project page, `http://www.4ward-project.eu`
3. Bertin, P., Aguiar, R., Folke, M., Schefczik, P., Zhang, X.: Paths to Mobility Support in the Future Internet., ICT-MobileSummit, Santander, Spain (June 2009)
4. Van Jacobson, K.: Introduction to Content Centric Networking. In: FISS 2009, Bremen, Germany (June 2009)
5. Ford, B., Iyengar, J.: Breaking Up the Transport Logjam. In: Proc. ACM HotNets-VII (October 2008)
6. D-5.2.0, Mechanisms for Generic Paths, public 4WARD deliverable (2009), `http://www.4ward-project.eu/index.php?s=file_download&id=38`
7. Day, J.: Patterns in network architecture: a return to fundamentals (Bd. 1). Prentice Hall International, Englewood Cliffs (2008)
8. OMNeT++ Discrete Event Simulation System Version 4.0