article published in

IEEE Wireless Communications Magazine
Special Issue: "Composite Reconfigurable Radio Networks"

June 2006

download: http://www.comsoc.org/livepubs/pci/public/2006/jun/index.html

Title:

Efficient Multi-Mode Protocol Architecture for Complementary Radio Interfaces in Relay-Based 4G Networks

Abstract: Wireless networks of next generation will satisfy the user demand for a ubiquitous mobile broadband access. Current research efforts are targeting therefore at the efficient realization of a flexible radio interface and network architecture. The wide range of application scenarios seems to prohibit a "one radio interface fits all" solution. Instead, it requires a multitude of solutions tailored for specific communication environments. Ideally, these solutions are related radio interface modes that have a common technology basis. This article presents a protocol reference architecture that enables the efficient integration of multiple such modes in a complementary way. This architecture essentially benefits from the commonalities between these modes. Hence it facilitates the coexistence and the cooperation of different modes in all devices of a future relay-based radio access network: (i) user terminals, (ii) access points and (iii) relay nodes. This article focuses on the realization of multi-mode capability of user and control planes and the related management of the protocols of a flexible radio interface. A proof of concept and its evaluation are presented at the example of relay-based 4G broadband networks and IEEE 802.16 (WiMAX).

Keywords: Multi-Mode Protocol, Radio Interface, Relay-Based 4G Networks, WiMAX, Wireless Broadband

Authors: Lars Berlemann, Ralf Pabst, Bernhard Walke

Affiliations: RWTH Aachen University, Faculty 6, Chair of Communication Networks (ComNets)

Addresses: Kopernikusstraße 16, D-52074 Aachen, Germany

Telephone: +49 241 80 27248

Fax: +49 241 80 22242

E-Mail: ber@comnets.rwth-aachen.de

Efficient Multi-Mode Protocol Architecture for Complementary Radio Interfaces in Relay-Based 4G Networks

Lars Berlemann, Ralf Pabst, Bernhard Walke
RWTH Aachen University, Faculty 6, Chair of Communication Networks
D-52074 Aachen, Germany
{ber|pab|walke}@comnets.rwth-aachen.de

***Abstract*— Wireless networks of next generation will satisfy the user demand for a ubiquitous mobile broadband access. Current research efforts are targeting therefore at the efficient realization of a flexible radio interface and network architecture. The wide range of application scenarios seems to prohibit a "one radio interface fits all" solution. Instead, it requires a multitude of solutions tailored for specific communication environments. Ideally, these solutions are related radio interface modes that have a common technology basis. This article presents a protocol reference architecture that enables the efficient integration of multiple such modes in a complementary way. This architecture essentially benefits from the commonalities between these modes. Hence it facilitates the coexistence and the cooperation of different modes in all devices of a future relay-based radio access network: (i) user terminals, (ii) access points and (iii) relay nodes. This article focuses on the realization of multi-mode capability of user and control planes and the related management of the protocols of a flexible radio interface. A proof of concept and its evaluation are presented at the example of relay-based 4G broadband networks and IEEE 802.16 (WiMAX).**

***Keywords*— Multi-Mode Protocol, Radio Interface, Relay-Based 4G Networks, WiMAX, Wireless Broadband**

I. Introduction

Future wireless networks will realize the vision of a ubiquitous radio system in providing wireless access, from short-range to wide-area, with one single adaptive system for all envisaged radio environments. This system will efficiently adapt to various scenarios by using different Radio Access Technologies (RATs), i.e., modes, with a common technology basis. The devices of a heterogeneous wireless infrastructure that use multiple modes (possibly simultaneously) can be referred to as *composite radios* [1]. The composite radio concept implies pre-installed modes complementing each other for optimized network utilization and Quality-of-Service (QoS) support. The dynamic selection, installation and adaptation of the devices' modes to the communication environment and changing user demands are introduced with the *reconfigurable radio* concept [2] based on the idea of software defined radios [3].

The integrated projects WINNER and E^2R funded by the European Union concentrate both on aspects of the vision introduced above in complementing each other: Relay-based wireless mobile broadband systems [4] are developed in WINNER as a promising candidate for wireless networks of the fourth generation (4G). Corresponding to the definition from above, such a relay-based system can be regarded as a *composite network* when multiple RATs are designated for its operation as outlined in Section II. The different RATs of the WINNER system are referred to as modes. The modes define similar radio interfaces tailored for specific communication environments. The efficient realization of different WINNER modes in a single device requires adaptability and touches thus the aspect of reconfigurability. Reconfigurable devices and supporting system functions such as reconfiguration management and over-the-air download of software components are the key objective of E^2R.

This article focuses on the multi-mode reference architecture for a flexible 4G radio interface. Therefore, the efficient realization of multi-mode protocols with the help of a generic protocol stack is introduced in Section III. The separation of protocol software into generic and specific parts is the basis for the multi-mode protocol reference architecture discussed in Section IV. Optimized switching between modes and coexistence of different modes is realized with the help of a hierarchical management structure on the level (i) of the complete protocol stack and of (ii) a single layer as discussed in Section V. Additionally, the realization of a flexible radio interface based on generic protocol functions of the Data Link Layer (DLL) is illustrated. A proof of concept and its evaluation are presented in Section VI at the example of relay-based 4G networks and IEEE 802.16 (WiMAX).

In the following, the term "multi-mode" may be equivalently substituted by "multi-standard", "multi-protocol" or "multi-system" and the term "layer" implicates a "sub-layer".

II. Composite Reconfigurable Radios in Relay-based 4G Networks

Relay-based wireless mobile broadband systems that provide a contiguous coverage in densely populated urban areas can serve as a prototypical example of potential 4G systems. Depending on the multi-mode capability of the network elements, different deployment scenarios can be identified as depicted in Fig. 1. The network elements related to the radio interface of a relay-based system are namely the User Terminal (UT), the Relay Node (RN) and the Access Point (AP). APs offer a high capacity in a relatively small coverage area. RNs have the advantage of distributing this high capacity available at the AP into a larger region. RNs cost-efficiently extend the coverage of a single AP to areas originally not covered by this AP. The capacity of the fixed link between AP and RN can benefit from antenna gain, spatial diversity and the exploitation of heavy shadowing through high spatial frequency re-use.

For the sake of simplicity we limit in our illustrations of Fig. 1 the available modes to two (here: 1 and 2). In the case of single-mode network elements the relay-based system has a classical multi-hop architecture as depicted in scenario (I.) of Fig. 1. In a wireless feeder scenario (II.), the RN simultaneously uses different modes of the radio interface for the RN-AP link and the UT-RN link respectively. These modes are preinstalled and used in a complementary way. Thus, corresponding to the definition from above, the RN can be regarded as a composite radio. Scenario (III.) additionally introduces a composite AP: The AP uses one mode for the relay link and a different mode for providing services to UTs. An additional third mode might exist to provide a highly reliable low bit-rate link between the AP and RN for signaling purposes (as for instance the Radio Resource Management (RRM)). The same stands for the AP-RN link in the composite radio network scenario (V.) where all network elements are multi-mode capable. Such a composite radio network allows an efficient fulfillment of capacity and QoS requirements and implies a joint RRM, inter-mode scheduling and self organization as discussed later. The reconfigurable network scenario (VI.) of dynamic configurable network elements with protocol reconfiguration and over-the-air software downloads leaves the scope of this article but is nevertheless facilitated through the multi-mode protocol reference architecture discussed below.

III. Generic Protocol Stack for Efficient Multi-Mode Capable Protocols

This section introduces the idea of a generic protocol stack for efficiently realizing multi-mode capability in communication protocol software on conceptual level. Its implementation is introduced in Section VI.

A. Rationale

The basic idea of a generic protocol stack is that all communication protocols share much functional commonality. This commonality can be exploited to build an efficient multi-mode capable wireless system. The term “generic” can be substituted in the following by “common” and “general”. Our key objective is to gather these common parts in a single generic stack. The generic parts are specialized in following particular requirements of a targeted mode (also referred to as RAT) and depicted in Fig. 2. The targeted advantages of this concept are: code/resource sharing and protocol development acceleration through reusability, maintainability and runtime reconfigurability.

A key rationale of the reference architecture for multi-mode protocols proposed in this article is the separation of a layer into specific and generic parts. The term generic is used by multiple authors with different knowledge backgrounds. This leads to dissimilar or even contradictory understandings of genericity. In taking the realization of a protocol stack out of generic and specific parts into account genericity becomes a software engineering problem of generic programming [5]. Generic programming can be defined as “programming with concepts”. A concept implies here a family of abstractions that all have a common set of requirements. The main effort in generic programming, especially in the design of generic software components, is the definition of these requirements. They have to be general enough to be met by a large family of abstractions but still restrictive enough to guarantee an efficient operation. Balancing the trade-off between general usability and implementation effort is therefore crucial for the success of separating complex protocol software into generic and specific parts.

As depicted in Fig. 2, generic parts can be identified in different aspects in the context of communication protocols:

- **Architecture and composition** of a protocol stack, as highlighted in Section IV
- **Functions** fulfilled by a layer that imply a certain behavior, as discussed in Section V
- **Data structures**: Protocol data units or information structures used for communication between peer-entities of a layer
- **Protocol framework**: Common rules for communication, as for instance the structure of a Medium Access Control (MAC)-frame (e.g. sequence and duration of broadcast, downlink and uplink phase)

- **Management** of a layer and protocol stack

These five characteristics of genericity form, together with mode-specific parts, a system specific protocol stack. An efficiently designed multi-mode capable stack is realized in adding management functions to take care of composition and parameterization of mode-independent and mode-specific parts. This ensures modes convergence, which is introduced in the next section in taking up the composite radio paradigm (in using modes in a complementary way). The introduction of additional reconfiguration management and related functions [6] leads to a dynamic reconfigurable protocol stack.

B. Related Work on Protocol Design

From the software engineering perspective, there are in general two possibilities to realize a generic protocol stack: (i) Parameterizable functional modules and/or (ii) inheritance, depending on the abstraction level of the identified protocol commonalities. The focus of this article is on the modular approach, as introduced in [7], while the inheritance-based approach is in considered in [8]. The combination of both approaches is promising to fulfill all requirements of protocol reconfigurability. A generic link layer for multi-radio transmission diversity and multi-radio multi-hop networking is discussed in [9]. This can be seen as a use case for the generic protocol stack elaborated in this article. However, not only the link layer protocols but also higher layer functions as for instance the control and management of the radio resources as well as mobility have to be considered in a multi-mode capable network. The concepts presented here are applied in [10] for a framework for implementing protocol layers. This approach is based on functional protocol units to efficiently support the design of a multi-mode protocol layer in the context of wireless communication. A similar concept is proposed in [11] for a non-layered, role-based design and implementation of fixed network protocols.

C. Separation of Protocol Stack into Generic and Specific Parts

The reference architecture presented in this section is based on the widespread perception that radio interface protocol functions can be divided into two sets of functionalities:

I. **Mode-/System-specific functions**: These are protocol functions that are unique to a certain kind of radio interface mode and can not be found in any other mode. Examples for such mode-specific functions are the allocation of a dedicated physical resource and parameters for adapting a mode to the local communication environment.

II. **Generic (common) functions**: In protocol standardization "generic" functions are not fully specified and have to be enriched by specifying missing parts. This view is extended here: In our context, the "generic" functions are assumed to be an identified set of common (mode independent) functions from a multitude of radio interface modes. This means they are "generic" from the viewpoint of the modes, but not from the viewpoint of functionality. These functions can be adapted to their utilization in any of the targeted modes through adequate parameterization. They will have to rely on additional mode-specific functions to provide the full functionality in a specific protocol layer of a specific radio interface mode. An Automatic Repeated reQuest (ARQ) protocol for instance, which provides an error-free data transfer, is a generic protocol that can be specified as Go-Back-N, Selective Reject or Hybrid ARQ protocol.

In regarding the generic protocol stack as *toolbox of common protocol function*s that can be used in specific protocol implementations the multitude of protocol functions considered as being generic is increased: It suffices that they can be found in at least two protocol modes to be counted as belonging to the toolbox of generic protocol functions. This means that the generic protocol stack, when covering more than two modes, also might contain some functions that are not used in all modes.

In general, the degree of commonalities between a set of different radio interface modes is decreasing with the number of modes being integrated. The design processes of new radio interfaces can highly benefit from taking the proposed reference architecture into account. This results into an increased degree of commonalities between these new radio interfaces and thus reduces development and device costs. Consequently, the degree of similarities can be improved, when considering multi-mode capability based on genericity during the development and standardization of new protocols compared to the common parts of existing protocols for wireless communication.

IV. Multi-Mode Reference Architecture Enabling Modes Convergence

A. Definition of Modes Convergence

A relay-based 4G radio access network as introduced above benefits from the multi-mode reference architecture proposed here. The reference architecture facilitates the coexistence and the cooperation of different modes in all network elements. This efficient integration of multiple such modes shall further be referred to as "Modes Convergence".

B. Multi-Mode Protocol Architecture

Fig. 3 illustrates the architecture of a multi-mode protocol stack of a flexible radio interface. Here, two radio interface modes (red and blue) are considered. The layer-by-layer separation into specific and generic parts enables the efficient realization of a protocol stack for multiple modes. The separation is the result of a design process, here referred to as cross-stack optimization. This optimization implies the identification and grouping of common (generic) functions. The generic parts, marked green in Fig. 3, of a layer can be identified on different levels, such as the Physical (PHY), MAC and Radio Link Control (RLC) as shown in the figure. The generic parts are reused in the different radio interface modes of the protocol stack. All generic parts together are regarded as generic protocol stack. The composition of a layer out of generic and specific parts is depicted in Fig. 3.

The composition and reconfiguration of a layer is performed by the (N)-Layer Modes Convergence Manager ((N)-MCM). The protocol modules of generic functions are exemplarily depicted in Fig. 3: Some of them are reused in a layer and/or additional functions are taken from the toolbox of common protocol functions from the generic protocol stack. The Radio Resource Control (RRC) on the control-plane and the RLC on the user-plane are generic to the layers located above. Each mode has an individual management-plane. The cross-stack management of different modes completes the reference architecture for multi-mode protocols. It connects the management-planes of the individual modes with the help of a Stack Mode Convergence Manager (Stack-MCM). The Stack-MCM and (N)-MCM exchange in a hierarchical order data between two modes. The switching between modes and the coexistence of several modes is performed by the Stack-MCM. The (N)-MCM enables the composition of a layer out of different parts as depicted in Fig. 3.

Concrete, the (N)-MCM manages a single layer and has the following tasks and responsibilities, which are introduced in the next section:

- **Layer composition and reconfiguration**: Arranging functionalities and setting mode-specific parameters
- **Protocol convergence**: Integrating different modes in one protocol layer, also allowing coexistence and cooperation of modes
- **Data preservation and context transfer:** Facilitating seamless switching between modes

Furthermore, the convergence between mode-specific protocol stacks is realized through the Stack-MCM implying implicitly the following functions:

- **Joint Radio Resource Management:** Radio resource coordination between different modes
- **Inter-mode scheduling**
- **Self-organization:** User data flow routing and autonomous forming of a network

The reconfiguration-plane located, in Fig. 3 besides the management-plane, is not considered in this article. The Stack-MCM realizes the reconfiguration of the protocol stack from one mode to the other and provides services to the reconfiguration plane. The reconfiguration plane contains all functions related to reconfiguration management [6] as for instance the security aspects of reconfiguration and software download as well as the communication of the reconfiguration capabilities of a device.

C. Functions of Stack Modes Convergence Manager

1) Joint Radio Resource Management

The functions of the user- and control-plane are administrated by the RRM. The RRM can be done centralized or decentralized and the RRM decisions are executed by the RRC of the corresponding modes. The RRM can assign multiple modes to one specific data flow. The RRC provides status information about the mode-specific protocol stack in a generic data structure to the RRM of the multi-mode protocol. In case of a (semi-) centralized coordination of radio resource allocation, this generic information structure about the status of the different modes of the protocol stack can be transmitted to the coordinating device for enabling an adequate decision. The RRM of a single multi-mode device may also support the coordination with neighboring devices as for instance a coordination across APs.

2) Inter-Mode Scheduling

The Stack-MCM, as intermediator between modes, performs inter-mode scheduling among different modes, as illustrated in Fig. 3. Contrary, the scheduling inside a mode between logical/transport channels is done in MAC-g or RLC-g of the mode's protocol layer. The inter-mode scheduling considers the dynamic scheduling of different user data flows over multiple modes. The scheduling strategy can for instance be based on the modes' interference situation. This requires a cross-stack optimization in providing information directly from the PHY the MAC or

RRC/RRM depending on where the decision is made. This information about the quality of the radio link is again provided in a generic information structure.

3) Self-Organization

The envisaged communication system is able to autonomously decide about its radio resource allocations under consideration of the local environment. This implies for instance the adequate selection of frequencies used for transmission or the forwarding of user data packets. The radio resource is selected under consideration of interference avoidance with other radio systems. Further, the optimized spectrum utilization coordinated with neighboring radio systems of the same technology is taken into account. This can also be related to efficient multi-hop relaying depending on the selected deployment scenario. The self-organization comprises plug-and-play scenarios of breaking down and installing additional devices in an operating communication system. The Stack-MCM has to support the addressed functionalities in activating for instance different modes to provide information about the interference situation or the topological role of devices in reception range (if it is acting as a RN or AP).

V. Modes Concergence in a Protocol Layer

The introduced hierarchical management structure differs between the complete stack and single protocol layer. The counterpart of the Stack-MCM (complete stack) is the (N)-MCM (single layer). It exists once in each layer and is introduced in this section.

A. Functions of (N)-Layer Modes Convergence Manager

1) Protocol Convergence

The convergence of multi-mode protocol stacks has two dimensions: First the convergence between two adjacent layers, in the following referred to as *vertical convergence* as it is known from the user-plane of the IEEE 802.16 protocol stack [12]. Second the convergence between layers located in the different modes of the protocol stack that have the same functions: In the following referred to as *horizontal convergence*. The generic protocol stack, managed by the (N)-MCM as introduced above, enables both the horizontal as well as vertical protocol convergence.

From the perspective of higher layer protocols the multi-mode protocol stack is transparent on the user- as well as on the control-plane as depicted in Fig. 4. Thus, generic parts terminate the stack towards higher layers. The vertical convergence of the (N)-MCM implies the adaptation of the packet data protocols to the specific mode, as for instance the conversion of an IP datagram in compressing the IP-Header.

2) Layer Composition and Reconfiguration

The separation into generic and specific parts requires an administration when taking the switching between modes into account: The common generic parts of the old mode need to be adapted for being reused in the new mode of the protocol stack. We assume that the generic parts of a layer exist permanently and are to be reconfigured and/or recomposed by the (N)-MCM corresponding to the characteristics of the targeted new mode. This assumption implies a module-based composition concept of the generic parts as introduced in [7] and [10]. The composition and configuration of the layer out of generic and specific parts is done by the (N)-MCM.

3) Data Preservation and Context Transfer

The communication between two modes and the mapping between generic and specific parts of a mode is done by the (N)-MCM (inside a layer) and the Stack-MCM (in case of a transfer between modes). The switching between two modes can be optimized in transferring the data from the old mode to the new mode. A user-plane protocol is able to reuse status information and protocol data in the generic part after switching to another mode. Therefore, an extension of the protocol into the control-plane is required although it performs only user-plane tasks. Depending on the status of the related protocol parts, the data transfer is referred to as *data preservation* or *context transfer* as illustrated in Fig. 3. If the generic part is reconfigured and recomposed the data needs to be preserved, i.e., adapted, to the new mode. In the case of a deletion of the old specific/generic part the data transfer is named context transfer. This implies data preservation for the new mode.

B. Composition of Layer from Specific and Generic Parts

Fig. 4 illustrates the general structure of a protocol layer conforming to our reference architecture from above. It is assumed that the functionality inside a layer is always composed out of generic and mode-specific parts. These parts jointly provide the modes' services of the layer via SAPs. Modes can also coexist temporarily or permanently. The specific SAPs of a layer are defined via the currently used mode or set of modes. This does not preclude the possibility that SAPs of different modes can be accessed by higher layer entities in a common way, as visualized by L(N) SAP-g.

The composition and reconfiguration of the layer is done by the layer-internal (N)-MCM, which resides in the management plane. It is the layer's counterpart of the Stack-MCM as introduced above. The (N)-MCM enables a (N)-layer to provide multiple modes and makes functionality of one or several modes available. Additionally, the (N)-MCM serves as a reconfiguration handler in each layer of the radio interface's protocols.

C. Generic Protocol Functions as Parameterizable Modules

The generic protocol stack implements common functions based on modules as introduced above. These common protocol functions get their system-specific behavior based on parameterization. Once specified, these modules can be repeatedly used with different sets of parameters reflecting a specific communication system. The modules of generic protocol functions form together with system-specific modules a complete protocol layer, as depicted in Fig. 4. The communication inside a layer is performed by employing generic service primitives and generic Protocol Data Units (PDUs) [7], which are also considered as being a part of the generic stack according to Fig. 2.

D. Generic Protocol Functions of Data Link Layer

As the architecture of most modern communication protocols cannot be entirely forced into the classical layered architecture of the ISO/OSI reference model, it is rather difficult to identify similarities and attribute these to specific layers. Therefore, this article deepens the level of examination in the search for similarities and considers fundamental protocol functions. Though these protocol functions mainly correspond to the DLL as defined in the ISO/OSI reference model, they can be found in multiple layers of today's protocol stacks [7]. The following functions of the DLL are considered as being part of the generic protocol stack:

- Error handling with the help of Forward Error Correction or ARQ protocols as for instance Send-and-Wait ARQ, Go-back-N ARQ or Selective-Reject ARQ
- Flow control
- Segmentation, concatenation and padding of PDUs
- Discarding of several times received segments
- Reordering of PDUs
- Multiplexing/De-Multiplexing of the data flow, as for instance the mapping of different channels
- Dynamic scheduling
- Ciphering
- Header compression

First examples for the composition of specific layers out of these modular protocol functions and the related performance evaluation were published in [7].

E. Parameterization of Functional Modules

Parameterization implies here not only specific values, as for instance the datagram size of a segmentation module, but also a configuration of behavior and characteristics of a module, as for instance the concretion of an ARQ module as a Go-back-N ARQ protocol with specified window sizes for transmission and reception. This implies as well a configuration of the modules' interface to the outside. The parameterization of functional modules can imply (i) a specification of certain variables, (ii) the switching on/off of certain functionality/behavior and (iii) an extension of the module's interface to the outside.

At the example of an ARQ module, the parameterization implies among other things:

- ARQ protocol characteristic, for instance Go-Back-N ARQ or Selective-Reject ARQ
- Transmitter or/and receiver role
- Receive and transmission window size
- Fixed, variable (TCP) window length or open/shut mechanism (GSM Logical Link Control (LLC))
- Timer value, after that a packet is assumed to be lost
- Connection Service: inexistent (UMTS RLC), separated for each direction (IEEE 802.11 - CSMA/CA with RTS/CTS), 2-way handshake (GSM LLC) or 3-way handshake (TCP)

- Use of Negative ACKnowledgments (NACKs)

Taking the example of the segmentation/reassembly module, the parameterization implies:

- Use of concatenation
- Use of padding, i.e., filling up of the PDU to reach a certain size
- Transmitter or/and receiver role
- Buffer size for Service Data Units concatenated in a single PDU
- Size of PDU after handling
- Behavior in case of error, i.e., interworking with ARQ module

The introduced ARQ module and its usage in different RATs is described and analyzed in detail in [7].

VI. Proof of Concept - Towards Convergence of IEEE 802.16 and WINNER Data Link Layer Protocols

In this section, we estimate our multi-mode architecture's capability of accommodating complementary radio interfaces. We consider a scenario where the Stack-MCM composes and configures the user-plane of the DLL for the operation in two modes. The RATs of the considered modes are (i) IEEE 802.16 (WiMAX) [12] and (ii) a relay-based 4G broadband system [4] as currently envisaged by the WINNER project. As above discussed, the WINNER system concept itself also envisages operation in different modes. These WINNER modes are mainly characterized through combinations of duplex- and multiple access-schemes and related frame timings. However, due to the strongly aligned design process (in applying our reference architecture) these modes already exhibit a high degree of commonalities, as reasoned in Section III. Therefore, two modes without a common design process were chosen here for our comparison and proof of concept.

Fig. 5 describes the composition of the WiMAX and WINNER DLL user-planes out of a set of mode independent and mode-specific functions according to the software framework[1] presented in [10]. The used functions are here further subdivided into two different classes according to Section III.C:

The **common, system-independent functions** are marked as light green boxes. They are taken from the toolbox of generic protocol functions presented in Section V. The **mode-specific functions** are colored as red and blue boxes respectively. For the two systems in this comparison case, the mode-specific functions include:

- Service Classification of outgoing data, where IP addresses and *Type of Service* have to be mapped on mode-specific connection or flow identifiers and QoS classes.
- System-specific handling of acknowledgement PDUs in the WiMAX DLL (AckSwitch).
- Service Level Control (SLC) in the WINNER DLL, dealing with, e.g. per-hop addressing, QoS control and selection of physical layer mode in the case of WINNER-internal multi-mode operation.
- Scheduling and timing functions that perform the actual mapping of data flows onto physical resources (in WINNER terminology: Resource Scheduler (RS)). It should be noted that it is possible to look at these functions at a higher level of granularity, again identifying commonalities, e.g. with regard to scheduling algorithms, as already mentioned in Section V.D.

The mode-specific functions are the only modules that definitely have to be replaced when a change between modes is performed. According to the requirements of the target mode, some of the mode-independent functions could be removed or added. Some functions may undergo a change of their parameter set, but their essential functionality remains the same. The change of the parameters is the responsibility of the (N)-MCM that manages the respective protocol layer (see Fig. 3).

When assessing the efficiency of this approach, the main parameters to be measured are (i) the degree of generic protocol functions taken from the toolbox and (ii) the degree of commonalities between different modes of operation. A simple and straightforward counting exercise yields that in the WiMAX case, 75% functions employed in the DLL are taken from the toolbox of generic protocol functions, while in the case of the WINNER radio interface's DLL, even 82% of the functional modules are generic, in case all optional functions are included. When concerning the

---

[1] The FlowSeparator function appearing in Fig. 5 is an implementation detail in the context of this software framework. It exceeds the scope of this article but is shown for the sake of completeness.

layered ARQ concept envisaged in WINNER, it is interesting to note that a reuse of functions is even possible inside one mode, which is another indicator for the implementation efficiency of the toolbox approach. This is also backed up by the number of protocol functions common to both modes (here at least 8). Investigations published in [7] have shown that protocols like the 802.11 DCF can also be composed from our toolbox of protocol functions with relative ease.

VII. Conclusion

The introduced multi-mode protocol reference architecture facilitates the structuring of an arbitrary layer into generic and specific parts. In providing guidance for understanding this structuring it marks up optimization potential in questioning the necessity of indicated differences. In this way, an increased protocol convergence is reached enabling an efficient multi-mode protocol stack for future wireless systems. The proposed reference architecture is applied in the development process of the WINNER system.

Relay-based wireless mobile broadband systems are set into the perspective of composite reconfigurable networks. Candidate functionalities of a flexible protocol stack are identified and a proof of concept including an evaluation is given. The introduced evaluation shows a high degree of function reuse and thus indicates the implementation efficiency of the proposed concept. The resulting flexibility has a deep impact on the design process and standardization of future communication protocols.

Acknowledgements

The authors would like to thank their colleagues M. Schinnenburg and C. Hoymann for the fruitful discussions and comments to the content of this article. Thanks also go to the European Commission-funded FP6 IST-Project WINNER (IST-2003-507581, http://www.ist-winner.org), which has partly funded the work presented. The authors would also like to acknowledge the inspiring comments of their colleagues in WINNER, although the views expressed are those of the authors and do not necessarily represent the project.

References

[1] P. Demestichas et al., “Management of Networks and Services in a Composite Radio Context,” in *IEEE Wireless Communications*, vol. 10, no. 4, Aug. 2003, pp. 44-51.

[2] P. Demestichas et al, “Evolution in Wireless Systems Management Concepts: From Composite Radio Environments to Reconfigurability,” in *IEEE Communicarions Magazine*, vol. 42, no. 5, May 2004, pp. 90-98.

[3] J. Mitola, “The Software Radio Architecture,” in *IEEE Communications Magazine*, vol. 33, no. 5, May 1995, pp. 26-38.

[4] R. Pabst et al., “Relay-Based Deployment Concepts for Wireless and Mobile Broadband Radio,” in *IEEE Communications Magazine*, vol. 42, no. 9, Sept. 2004, pp. 80-89.

[5] J. Gibbons, J. Jeuring, “Generic Programming,” Kluwer Academic Publishers, Boston USA, Feb. 2003.

[6] N. Alonistioti et al., “Reconfiguration Management Plane for the Support of Policy-based Network Reconfiguration,” in Proc. of *PIMRC’04*, Barcelona Spain, Sep. 2004.

[7] L. Berlemann, R. Pabst, and B. Walke, “Multi-Mode Communication Protocols Enabling Reconfigurable Radios,” in *EURASIP Journal in Wireless Communications and Networking, Special Issue on “Reconfigurable Radio for Future Generation Wireless Systems”*, vol. 2005, no. 3, pp. 390-400, Aug. 2005.

[8] M. Siebert and B. Walke, “Design of Generic and Adaptive Protocol Software (DGAPS),” in Proc. of *3Gwireless '01*, San Francisco USA, Jun. 2001.

[9] G. P. Koudouridis et al., ”Generic Link Layer Functionality for Multi-Radio Access Networks,” in Proc. of IST Mobile Summit 2005, Dresden, Germany, 19-23 Jun. 2005.

[10] M. Schinnenburg, F. Debus, A. Otyakmaz, L. Berlemann and R. Pabst, “A Framework for Reconfigurable Functions of a Multi-Mode Protocol,” in Proc. of Software Defined Radio Technical Conference 2005, Orange County CA, USA, 14-18 Nov. 2005.

[11] R. Braden, T. Faber, and M. Handley, “From Protocol Stack to Protocol Heap - Role-Based Architecture,” in *ACM SIGCOMM Computer Communication Review*, vol. 33, no. 1, Jan. 2003.

[12] C. Hoymann, “Analysis and Performance Evaluation of the OFDM-based Metropolitan Area Network IEEE 802.16,” in *Computer Networks*, vol. 49, no.3, pp. 341-363.

Biographies

LARS BERLEMANN [M] (ber@comnets.rwth-aachen.de) received his Dr. and Diploma degree in Electrical Engineering from RWTH Aachen University, Germany, in 2006 and 2002 respectively. He additionally holds a Diploma degree in Business and Economics from the same university. Since 2002 he serves as a research assistant at the Chair of Communication Networks of RWTH Aachen University, where he is now working as a senior research scientist at various national and European Union funded research projects. Dr. Berlemann has published more than 40 reviewed publications including 5 journal articles and was scientific organizer of European Wireless 2005 and

PIMRC 2005. His current research interests include the IEEE 802 standard family, coexistence, spectrum sharing, cognitive radio and flexible protocol stacks.

RALF PABST [StM] (pab@comnets.rwth-aachen.de) received his Diploma degree in Electrical Engineering from RWTH Aachen University, Germany, in 2001 and serves since then as a research assistant at the Chair of Communication Networks of RWTH Aachen University, where he is working towards his Dr. degree. He has been working on various national and European Union funded research projects and has been actively involved in the organization of the Working Group 4 (New Technologies) in the Wireless World Research Forum. His research interests include the performance evaluation of relay-based deployment concepts for future generation wireless networks, associated resource management protocols and efficient design of multi-mode protocols. He published over 25 scientific publications, including 3 journal articles and 2 Textbook chapters.

BERNHARD H. WALKE [SM] (walke@comnets.rwth-aachen.de) is running the Chair for Communication Networks at RWTH Aachen University, Germany, where about 30 researchers work on topics like air-interface design, development of tools for stochastic event driven simulation and analytical performance evaluation of services and protocols of XG wireless systems. Most of this work continuously has been funded from 3rd parties' grants. He is author of the 2002 book “Mobile Radio Networks – Networking, Protocols and Traffic Performance” and co-author of the 2001 book UMTS – The Fundamentals and the forthcoming 2006 book “IEEE 802 Wireless LAN/PAN/MAN Systems: Standards, Models and Traffic Performance”. He has been a board member of ITG/VDE and is Senior Member of IEEE. He has served as Program Committee and Steering Committee Chair of various conferences like the European Wireless (EW) conference that he co-founded. In 2005 he was the Scientific Chair of IEEE-PIMRC 2005, Berlin. His group has substantially contributed to the development of standards like ETSI/GPRS, ETSI/BRAN HiperLAN2, CEN TC 278 DSRC (electronic fee collection), IEEE 802.11e, 802.16 and 802.15.3. From 2001 to 2003 he was an elected Chair of Working Group 4 (New Technologies) of the Wireless World Research Forum. Prior to joining academia, he worked for 18 years in various industry positions at AEG Telefunken (now EADS AG). He holds a Dr. (1975) degree in Information Engineering from University of Stuttgart, Germany.

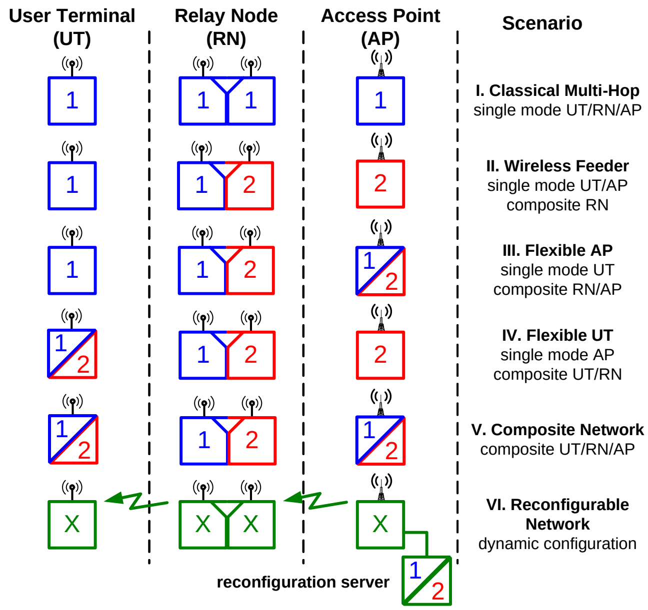

Figure 1. Deployment scenarios of relay-based wireless mobile broadband systems characterized through the different usage of two radio interface modes (here: 1 and 2).

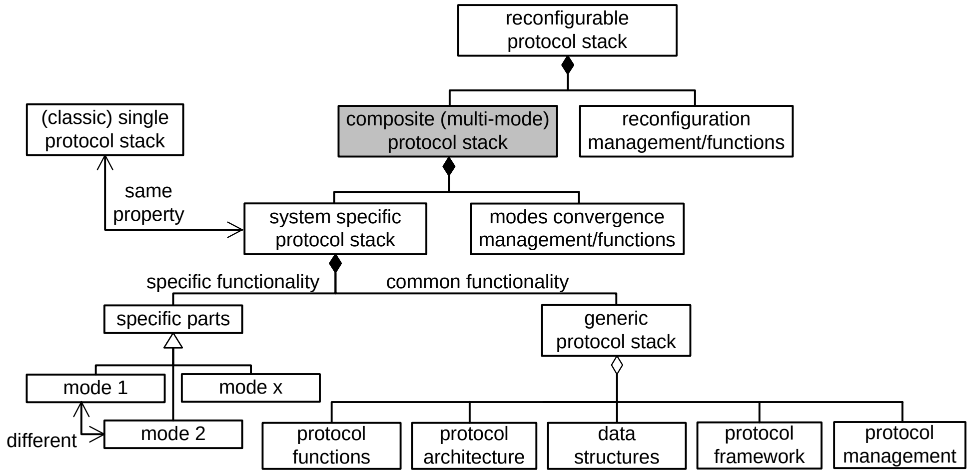

Figure 2. UML diagram of the generic protocol stack in the context of protocol multi-mode capability and reconfigurablility.

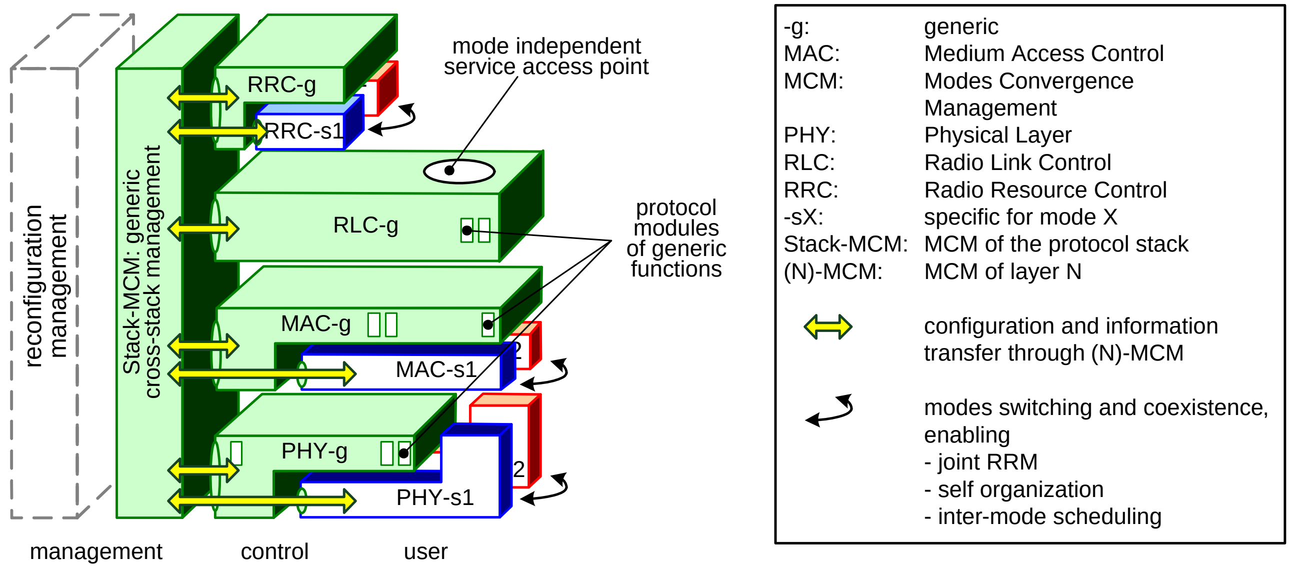

Figure 3. The multi-mode protocol reference architecture, facilitating switching between modes (inter-mode handover) and coexistence of modes (relay node connecting different modes). A cross-stack management is supported by the Modes Convergence Manager of a layer.

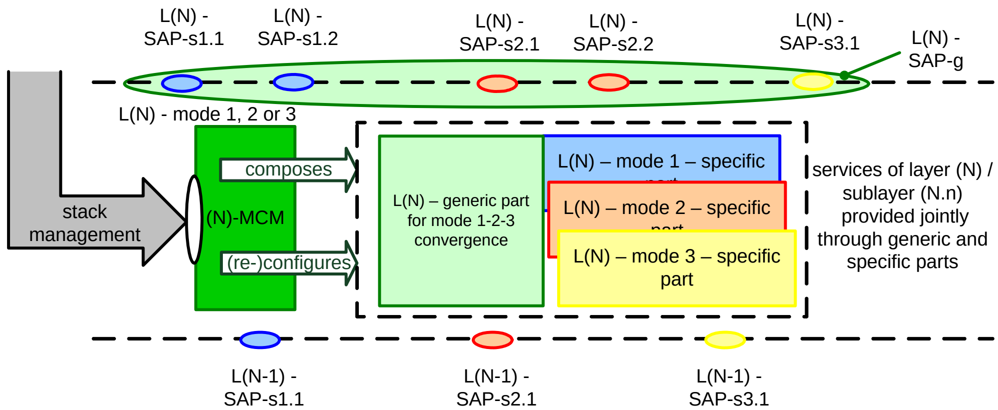

Figure 4. Composition of a layer (N) from generic and specific functions. The composition and (re-)configuration is handled by the (N)-MCM. The (N)-MCM is controlled by a layer-external stack management entity, namely the Stack-MCM.

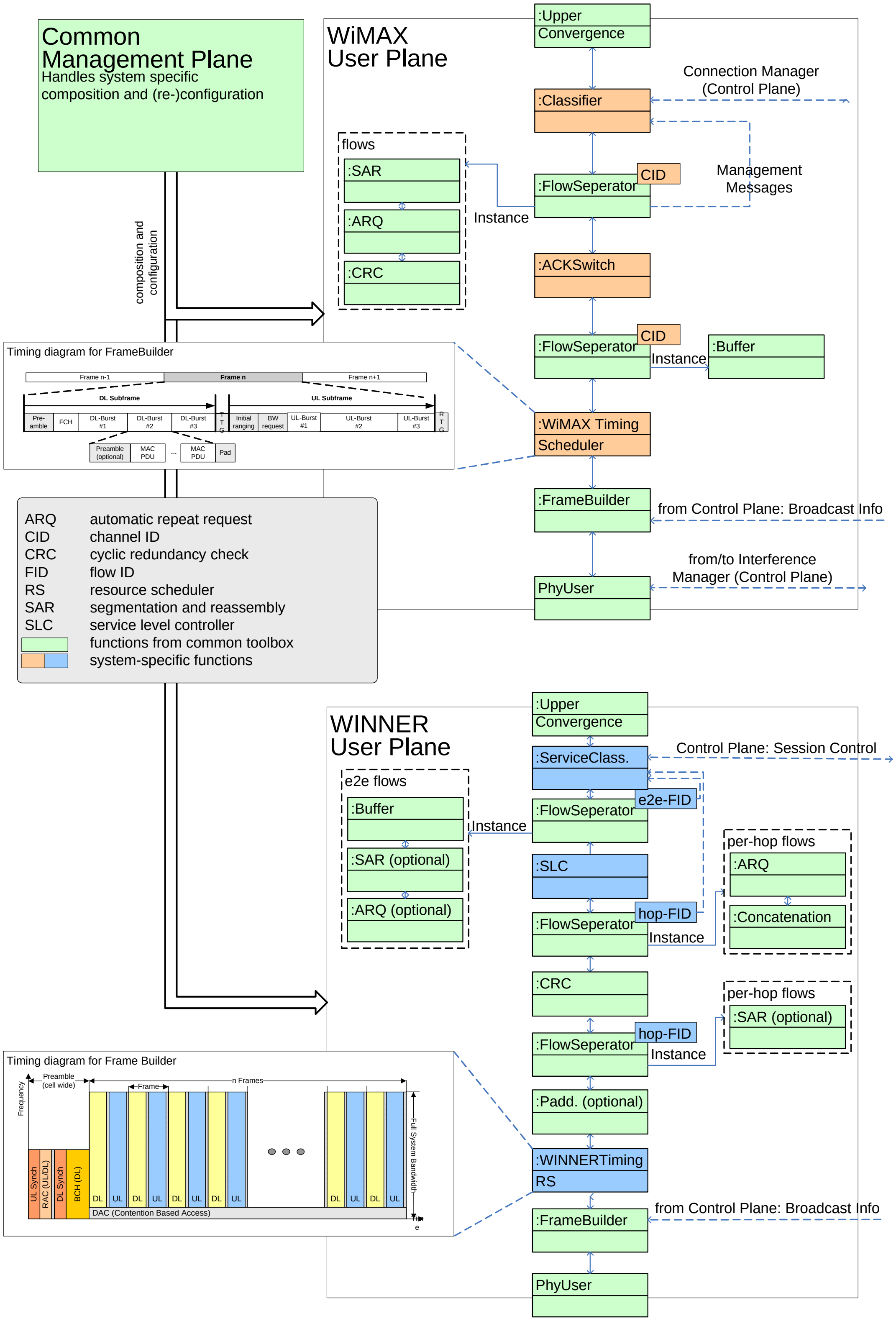

Figure 5. Proof of concept: Convergence of IEEE 802.16 (WiMAX) and WINNER data link layer protocols.