A Dynamically Extensible Control Space for Next-Generation Networks

Bengt Ahlgren, Lars Eggert, Norbert Niebert, Börje Ohlman, Christian Prehofer and Mikhail Smirnov

***Abstract*—Network interworking in the IP era is largely an issue of compatibility of control layer functions and protocols. No common framework exists yet which would tie together the various mechanisms established for achieving QoS, security, mobility, etc. This paper introduces the concept of an extensible control space that can dynamically incorporate existing or new mobile networks. The wide variety of different network functionalities in today's networks motivates this concept; heterogeneity in future networks will increase further. Providing a common, shared control space designed with a strong and consistent architectural view provides flexibility and dynamic adaptability of control functionality. Special emphasis lies on novel features to allow networks to automatically and dynamically interconnect and reconfigure their control functionality – which we call *network composition*.**

***Index Terms*— network interworking, heterogeneous systems, Ambient Networks**

I. Introduction

Ambient Networks is a large-scale collaborative project within the European Union 6th Framework Program that investigates future communications systems beyond today's fixed and 3rd generation mobile networks. It is part of the Wireless World Initiative [3]. We aim for a new concept called Ambient Networking, to provide suitable mobile networking technology for the future mobile and wireless communications environment. Ambient Networking will provide a unified networking concept that can adapt to the very heterogeneous environment of different radio technologies and service and network environments. Special focus is put on facilitating both competition and co-operation of various market players by defining interfaces, which allow the instant negotiation of agreements. This approach goes clearly beyond interworking of well-defined protocols and is expected to have a long-term effect on the business landscape in the Wireless World. Central to the project is the concept of *composition* of networks, which is our approach to address the dynamic nature of the target environment. The approach is based on an open framework for network control functionality, which can be extended with new capabilities as well as operating over existing connectivity infrastructure.

This paper first provides a brief overview of the needs for a new concept and the overall Ambient Networks architecture, which is further detailed by Niebert et al [4]. The architecture exposes three major interfaces, which are explained in more detail. Two particular aspects are expanded further, the naming framework and the connectivity abstractions. Internal mechanisms of the Ambient Control Space are highlighted, especially the control communication aspects, the control space registry and consistency control. The paper ends with a conclusion and outlook for further work.

II. The Need for a new Mobile Networking Concept

The mobile communications environment is changing. In the business environment, we see the emergence of a more complex value chain of players, each focusing on particular activities such as service creation, marketing, or infrastructure operation. Another change is the emergence of new radio access networks. No single radio technology is able to deal with all environments and usage scenarios in a scalable and affordable manner. The key to success lies in the efficient combination of many new and legacy radio resources.

Furthermore, the mobile networking world is extending outside the operator domain. In the enterprise and in the home, as well as in vehicles and in personal area networks we see the usage of wireless networking increasing. Finally, market success will depend on the competitive provisioning of new services tailored to the desires of users. Service providers have a need for an access agnostic network layer that enables them to create services quickly, economically and ubiquitously.

Current network technology is heading towards universal use of IP technology but is still not able to meet all the challenges that the future mobile environment imposes. For instance, new mobile network solutions are needed to cope with ever changing configurations in ad-hoc mobile networks and personal area networks. New scaleable concepts for

This paper describes work undertaken in the Ambient Networks project, which is part of the EU's IST programme. In total, 41 organizations from Europe, Canada, Australia and Japan are involved in this project, which will run from 2004-2005 in its first phase. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the Ambient Networks project. Bengt Ahlgren is also partly supported by the SSF funded Winternet research program. Lars Eggert is with NEC Network Laboratories, Heidelberg, Germany, (phone: +49-6221-905-1143; fax: +49-6221-905-1155; e-mail: lars.eggert@ netlab.nec.de). Börje Ohlman is with Ericsson Research, Kista, Stockholm, Sweden, (phone: +46-8-508 78095; fax: +46-70-616 3187; e-mail: Borje.Ohlman@ericsson.com). Norbert Niebert is with Ericsson Research, Aachen, Germany (e-mail: Norbert.Niebert@ericsson.com). Christian Prehofer is with DoCoMo Euro-Labs, Germany, Prehofer@docomolab-euro.com, Mikhail Smirnov is with Fraunhofer FOKUS, Berlin, Germany (e-mail: smirnow@fokus.fraunhofer.de). Bengt Ahlgren is with SICS, Swedish Institute of Computer Science (http://www.sics.se/)

instant roaming agreements are needed to enable roaming between cellular networks and the increasing number of hotspot and privately owned mobile networks. These changes and challenges indicate that there is a need for a new mobile networking solution; a need for the new ambient networking concept that is outlined in this paper.

The Ambient Networks project has adopted a unifying principle to address these challenges: this principle is that future networking requirements should be addressed by developing the concept of a new type of network-level "building block", existing above the level of individual devices or functions. The building block must be flexible, to adapt to different specialised types of access and operational models; it must be possible to interconnect in a uniform manner, so that arbitrary combinations of technologies and business environments can be merged seamlessly; and it must be possible to carry out this connection simply and dynamically, fostering cooperation and competition, and openness to new business opportunities. This building block is the *Ambient Network* – a set of one or more nodes and/or devices, which share a common control plane (called the *Ambient Control Space*), and which implements well-defined external interfaces to Users or other Ambient Networks.

III. Ambient Networks Overview

Figure 1 shows a simplified overview of the logical structure of the *Ambient Control Space* (ACS.) It illustrates that an Ambient Network consists of three distinct components. First, the underlying *Ambient Connectivity*, an abstraction of existing network infrastructure controlled by the second component, the *Ambient Control Space*, which itself consists of two kinds of components: the actual control functions (the boxes in the control space) and the control space framework functions (not explicitly shown; implemented by the ellipse surrounding the connectivity plane.) The control space framework comprises all functions necessary to allow the control functions to plug into the control space, execute their control tasks and coordinate with other functions present in the control space. The actual control functions, such as

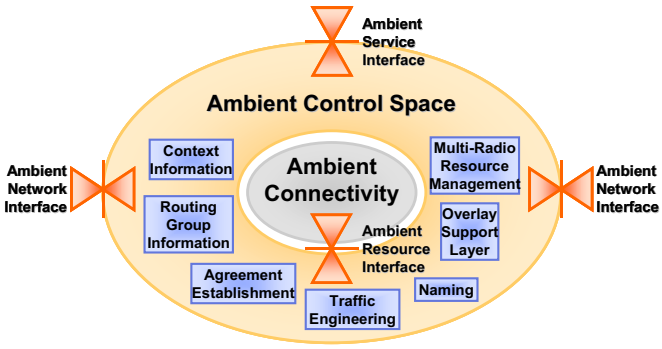

Figure 1: Illustration of the logical organisation of the Ambient Control Space.

overlay support, can be added in a plug&play fashion, based on the functionality of the control space.

A. Control space interfaces

The third and final component is the set of three control space interfaces: the *Ambient Resource Interface* for communication with connectivity resources, the *Ambient Service Interface* for interaction with services and applications and the *Ambient Network Interface* for communication with other networks.

The Ambient Network Interface (ANI) connects the control spaces of different Ambient Networks. The ANI is used for negotiation of network composition agreements and for transferring control information between the networks. The interface does not exist on every node of the network, but rather the nodes that collectively implement the core control space functionality.

The Ambient Service Interface (ASI) is located between the control space and the application inside a node. It allows applications and services to issue requests to the control space concerning the establishment, maintenance and termination of end-to-end connectivity between functional instances connecting to the ASI. The ASI also might include management capabilities and the means to make network context information available to the applications. The control functionality that the Ambient Control Space provides and exposes through the Ambient Service Interface makes it possible to implement services that are independent from specific underlying connectivity networks.

The Ambient Resource Interface (ARI) is located inside a node between the control space and the connectivity layer. It offers control mechanisms that the ACS can use to manage the resources residing in the connectivity plane. These resources can be routers, switches, radio equipment but also media transcoders, filters and proxies. The ARI shields the control space itself from the heterogeneity of the underlying connectivity networks and allows it to provide a common control layer.

B. Control space mapping to nodes

Figure 2 shows a logical view of two Ambient Networks. It illustrates that there is one common control space for all the nodes within an Ambient Network. The control space makes decisions on behalf of the nodes belonging to the network and controls some aspects of their operation. The control space is therefore logically present at each node. (The Ambient Network architecture does not mandate a certain kind of implementation of the control space; it can be implemented in centralized or in a distributed fashions.)

Nodes may implement parts of this distributed control space. For communication with other nodes in the same network that may implement other parts of the control space, a message passing mechanism for intra-control-space communication is used.

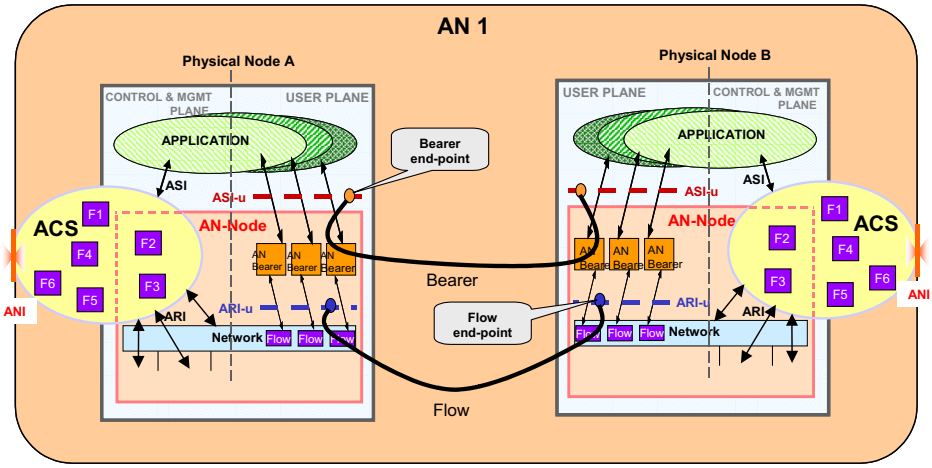

Figure 2: Example showing Ambient Network nodes and interfaces

Figure 2 also illustrates the relationships between the ASI, ARI, application programs and the interactions with the control space. Mapping these abstract control space concepts onto a physical node is not trivial. The two nodes in Figure 2 implement pieces of both the control and user planes. The picture also illustrates that the ASI and ARI are node-internal interfaces that are involved in both control and user traffic exchanges. Although logically separate, these two instantiations of the interfaces are expected to be implemented in a shared fashion.

Before describing the internals of the ACS in detail, the next section discusses the naming principles of the Ambient Network architecture as a prerequisite.

C. *Naming framework*

The Ambient Network naming framework focuses on supporting four key aspects of the overall Ambient Network architecture: global reachability across addressing domains, support for different services and end-nodes including control of intermediaries (middleboxes), mobility of services, nodes, networks and sessions/flows and resistance to security threats such as denial of service attacks or intrusions. A layered naming model can address these requirements [5][6][7].

The following four naming layers allow dynamic bindings between adjacent layers that enable native support for mobility of nodes and services. *Application services* or *data objects* have identities that are persistent over time and not tied to the end-system hosting the service or data. Examples are SIP services and web pages. *Application points of attachment* define the point where an application program implementing (parts of) an application service is reachable for clients. They are located at the ASI and can be compared to a standard TCP/IP socket API. *Host end*-systems are nodes in the network whose identity stays the same, regardless of their current location and communication interface. A host end-system does not necessarily denote a physical box – it may be a logical entity that can move between physical boxes. It is the entity that is hosting the ASI and ARI interfaces. Finally, *points of network attachment* are locations in the network that are identified by some kind of network addresses (also called *locators*.) Locators are often dependent on network topology and are defined on the ARI level.

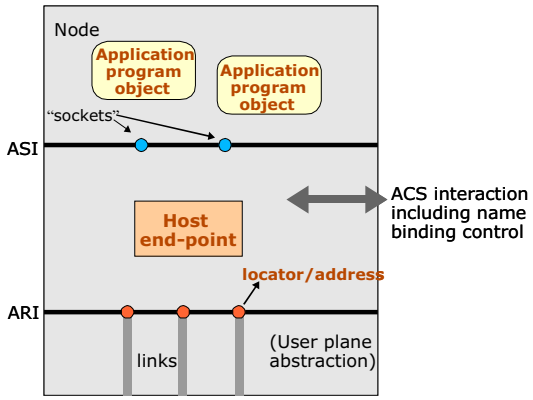

Figure 3: Named objects in the ACS

The purpose of defining a layered naming framework is to provide *dynamic bindings* between the levels. With dynamic bindings at multiple levels, names of objects become location-independent and different types of mobility, e.g., for nodes and services, become possible without add-on mechanisms.

D. *Connectivity Abstractions*

There are three levels of abstractions for connectivity in AN.

Session – An application specific notion of connectivity that we leave to the particular application to define the precise meaning in terms of mapping to bearer(s.)

Bearer – The connectivity abstraction that the Ambient Network provides to the application at the ASI.

Flow – An abstraction of the basic connectivity provided by the underlying network technology at the ARI.

We describe the latter two in more detail in the following two subsections.

1) *Flow*

A flow is an abstract view of the connectivity provided by the underlying network technology. Depending on the latter, a "shim" layer may be needed in between to adapt the technology to the abstraction. A flow is constrained to a single network technology.

A flow is a *transfer of data* between two instances of the ARI, where a technology dependent locator labels each flow endpoint. Flows are *unidirectional*, so a flow is associated (minimally) with a specific source locator and destination locator. For some types of network technologies, a flow may require a connection set up, but for others it is not necessary.

A flow may pass through *intermediate resources*, which are not explicitly tied to the flow, but which can be controlled through the ARI. The set of intermediates may change over the lifetime of the flow without changing the flow itself. The flow may also pass other nodes not visible, and thus not controllable, through the ARI.

2) Bearer

A *bearer* runs end-to-end between application peers. It is the means for communication that an Ambient Network provides to applications at the ASI. The bearer, unlike the flow, is not bound to locators, but to a higher-level object in the naming framework. This means that the bearer can make use of the functionality provided by the control space, such as mobility, address translation and media adaptation. For the latter, the bearer has (optional) media properties that tell the data manipulation functions of the control space what things are allowed or requested to be done with it.

For certain applications, e.g., a file transfer, a bearer can be quite simple requiring very little above what a flow provides. For other applications, e.g., voice, the bearer can be quite complex involving transcoding and special media routing.

The rest of this paper describes the common parts of the ACS in some detail by focusing on the underlying control space functionality, namely, control communication via message passing, a common registry and consistency mechanisms. These mechanisms are the basis to enable plug-and-play extensibility of the controls space, by adding more components that provide additional functionality.

Note that although the following sections describe message passing, registry and consistency control functionalities separately, these functions are interdependent. For example, consistency control is involved in concurrent updates to the registry and for some operations; direct message based-communication may be replaced with indirect communication through registry updates and queries. The details of these interdependencies will be investigated during the detailed functional specification of the architecture.

IV. Control Communication

Different functions within the ACS communicate by exchanging messages with one another. Message-based communication among a set of participants requires a number of globally agreed-upon principles. Participants need unique identifiers to enable unambiguous message delivery. A resolution mechanism must map these identifiers into locators for the specific message delivery mechanism. Two communicating parties must agree on a specific encoding for the information they transfer. Finally, the message passing service may need to implement additional services other than best effort delivery, such as guaranteed delivery, duplication prevention, reordering protection, prioritization, subscription or flow control, to support the particular communication needs of the participants.

These required features for message-based communication within the ACS are very similar to what the Ambient Networks user plane abstraction provides. In some sense, the ACS can be seen as a distributed application or service implemented on top of the generic user plane. Because the ACS is being designed within connectivity abstractions that provide a uniform view on specific user plane technologies, using the same communication mechanism within the control space offers considerable synergies: No additional communication system on top of the user plane abstraction is needed to support the control space, leading to a relatively thin interface towards the connectivity resources (ARI.) The remainder of this section will discuss how the generic user plane supports message-based communication within the control space.

First, message-based communication within the ACS requires the dynamic allocation, de-allocation and management of unique identifiers for individual control space functions. The naming functions for the generic user plane already support these operations. Similarly, binding identifiers to topological locators is also a key characteristic of the existing naming functionality, which the ACS can leverage. However, providing plug-and-play extensibility to the ACS likely requires specific further registry functionality.

Second, communicating parties must agree on a specific encoding for the information they transfer. This capability is *not* part of the generic user plane abstraction. Information encoding is a service-specific issue and must hence be addressed at the control space level. Information encoding for control space messages, especially extensible mechanisms that can incorporate new types of data, such as network level context information is currently an open issue under investigation. However, existing encoding schemes such as MIME, XML or ASN.1 may be readily adaptable.

Third, if a function receives conflicting information about global data, a consistency control mechanism must resolve the conflict. For long-lived, critical information, a system-wide agreement has to be established in case of such conflicts. This functionality is not part of the generic user plane and is currently being investigated within Ambient Networks. Section VI discusses the need for a consistency control mechanisms.

Finally, the generic user plane abstraction only provides a simple, best effort delivery mechanism for messages. Although this allows the connectivity abstraction to incorporate many different network technologies, for communication within the control space, best effort delivery may be too limited. A richer set of communication primitives, for example, guaranteed delivery, duplication prevention, reordering protection, prioritization, subscription or flow control, can provide improved communication mechanisms that simplify the implementation of control space functions by factoring out communication primitives into a common

substrate.

V. Registry

The registry is an ACS-wide directory and storage service accessible by all functions. In a very general sense, it is a distributed database. Providing a unified registry simplifies many control functions by factoring out storage, discovery, lookup, sharing, distribution and access control to information into a common service. The registry is logically a single service, but implementation of registry access and data storage is expected to be distributed for sizable Ambient Networks.

One purpose of the ACS registry is storage of information about user plane entities such as network resources, services, specific hardware, links, sessions, policies and user information that are used by functions in the control space. It controls access to this information, coordinates distributed use and manages persistent storage.

A second purpose of the ACS registry is storage of information about the ACS itself. In this function, the registry supports the message passing (section IV) and consistency control functionalities (section VI) within the ACS. For example, the ACS registry may maintain the bindings of ACS functions to topological locators.

In addition to these typical directory services, the ACS has to provide basic resource control functions to coordinate the different ACS components. This includes managing access to resources, which are identified in the registry. While some of the resource access control for specific entities can be provided locally by other components, some basic services will be required in the central repository. It is an item for further study which access control has to be done in an ACS wide repository. It also includes managing potential conflicts for the registry, e.g. if different entities aim to insert conflicting information. This issue will be discussed below

Similar to the message passing functions, data encoding is a challenge when designing the ACS registry. Due to the dynamic nature of the control space, the registry may accommodate many different kinds of information with potentially very different access characteristics.

There is also a context coordination function for Ambient Networks [8], which includes a context information base (CIB.) This database might already provide the required functionality or at least serve as a basis for the ACS registry. To what extend the CIB can be used to implement the registry function is still an issue being investigated.

Another Ambient Network specific capability of the registry is related to network composition. Network composition needs to combine the registries of two networks in an efficient and flexible way. When this is done the information in the composed registries must be checked for consistency.

VI. Consistency Control

Consistency control and conflict resolution mechanisms coordinate the concurrent, distributed decisions made by individual ACS functions. Inconsistencies can arise due to concurrent, conflicting updates to shared state, e.g., when the quality-of-service functions decide to initiate a handover to improve service quality, whereas power management decides to not initiate a handover to conserve power. Furthermore, inconsistencies can arise when a decision by a control function conflicts with policies established for an Ambient Networks by users, operators, or as part of a composition agreement.

The Ambient Network approach is to control consistency at different levels. This means that conflicts are best resolved where they are detected, i.e. at the level of an ACS function; if this is not possible, resolution within a functional area is used. Finally, at the level of the Ambient Control Space, generic conflict resolution functionality computes and maintains fairness relations between functions.

VII. Conclusion

This paper has presented an overview of the Ambient Networks architecture, focusing on several key functions of the control space and associated connectivity abstractions. Early results indicate the scalability and usefulness of the approach in a heterogeneous and dynamically changing network environment. Further work will focus on detailing the control space and specifying and integrating its functionality.

Acknowledgment

The authors would like to thank their colleagues in the Ambient Networks projects for many contributions, hints and discussions that have helped to shape the outline of the Ambient Control Space.

References

[1] IST project 507134 Ambient Networks, http://www.ambient-networks.org

[2] Ambient Networks – Research for Communication Networks Beyond 3G", N. Niebert et al., Proceedings of IST Mobile Summit, Lyon, France, June 27-30, 2004

[3] The Wireless World Initiative: A Framework for Research on Systems Beyond 3G", N. Niebert et al., Proceedings of IST Mobile Summit, Lyon, France, June 27-30, 2004

[4] Ambient Networks: A Framework for Future Wireless Internetworking", N.Niebert et al., Proc IEEE VTC 2005 spring, Stockholm, May 2005

[5] A Layered Naming Architecture for the Internet", H. Balakrishnan et al., Proc ACM SIGCOMM 2004, Portland, Oregon, USA, August 30 - September 3, 2004.

[6] Host Identity Protocol Architecture", Robert Moskowitz, draft-moskowitz-hip-arch-06 (work in progress), June 2004.

[7] "On the naming and binding of network destinations", Jerome Saltzer, In P. Ravasio et al., editor, Local Computer Networks, pages 311-317. North-Holland Publishing Company, Amsterdam, 1982. Reprinted as RFC 1498, August 1993.

[8] "Contextware Research Challenges in Ambient Networks", Ahmed Karmouch, e.a.. In. Proceedings of the Workshop on Mobile Aware Technologies and Applications MATA 2004 first international workshop, Florianâopolis, Brazil, October 20 - 22, 2004 ; proceedings / Ahmed Karmouch ... (eds.). - Berlin [u.a.] : Springer, 2004. - XII, 382 S. - (Lecture notes in computer science ; 3284) ISBN 3-540-23423-3