Ambient Networks: A Framework for Future Wireless Internetworking

Norbert Niebert
Mikael Prytz
Andreas Schieder
Ericsson Research
{norbert.niebert, mikael.prytz,
andreas.schieder}@ericsson.com

Nick Papadoglou
Network Technologies, Group R&D
Vodafone Group Services Limited
nick.papadoglou@vodafone.com

Lars Eggert
NEC Europe Ltd., Network Laboratories
lars.eggert@netlab.nec.de

Frank Pittmann
Siemens AG, New Technologies – Research
frank.pittmann@siemens.com

Christian Prehofer
DoCoMo Euro-Labs
prehofer@docomolab-euro.com

Abstract — **An increasingly wireless world faces new challenges due to the dynamicity of interactions, range of applications, multitude of available radio access technologies and network functionality. The *Ambient Networks* project recognizes these trends and enables the creation of innovative network solutions for mobile and wireless systems beyond 3G. These networks will enable scalable and affordable wireless networking while providing pervasive, rich and easy-to-use communication. A specific focus lies on enabling advanced capabilities in environments with increased competition as well as cooperation, environments that are populated by a multitude of user devices, wireless technologies, network operators and business actors. The project adopts a modular architecture that enables plug-and-play control extensibility that supports a wide range of different applications and network technologies. Based on a small subset of common functionality, this approach supports the dynamic deployment of advanced internetworking capabilities, such as media- and context-awareness or multi-radio access.**

Introduction

One of the main goals of the *Ambient Networks* project [1][2], part of the EU's 6th Framework Programme, is to enable seamless interoperation between heterogeneous networks. Ambient Networks aim to establish this interoperation through a common control plane distributed across the individual, heterogeneous networks. This new control plane functionality can be deployed both as an integral component of future network architectures that have better intrinsic support for network heterogeneity or as an add-on to existing, legacy networks that allows them to interoperate with future networks. In the latter case, this common control plane functionality "wraps around" legacy control functionalities, encapsulating and abstracting their individual control idiosyncrasies to provide at least the required subset for future interoperation. This well-defined migration path is first-order priority of the project and enables already deployed, legacy infrastructure to participate in the advanced internetworking capabilities provided by the Ambient Networks architecture.

Figure 1 illustrates the logical organization of the Ambient Control Space (ACS) internals, showing the functionality and interfaces that are its main features. The control space (large oval in Figure 1) consists of a collection of control functions, such as *naming* or *composition agreement* that cooperate to implement specific control functionality. These control functions exist within the overall control space framework.

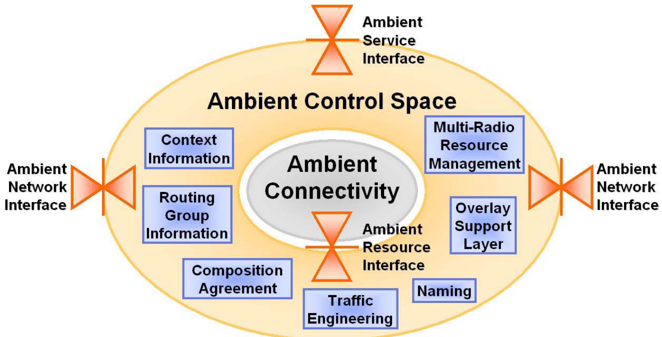
Ambient Service Interface
Ambient Control Space
Context Information
Multi-Radio Resource Management
Ambient Network Interface
Ambient Connectivity
Ambient Network Interface
Routing Group Information
Overlay Support Layer
Ambient Resource Interface
Composition Agreement
Naming
Traffic Engineering

Figure 1: Control space modularization and interfaces.

This paper introduces the overall architecture and several key functions of our new approach. First, it provides more details on the the interfaces of the control space and discusses some of its key components in detail. The control space integrates heterogeneous network technologies with different capabilities. Therefore, it must rely only on basic networking primitives, which it encapsulates in a set of connectivity abstractions. The control space further provides a common infrastructure for internal message passing, conflict handling, registry, and a second set of connectivity abstractions for the service layer. It also provides a framework that enables the simple inclusion of advanced networking functionalities. One such function that is expected to be present in many control spaces is a novel architecture for multi-radio resource management, presented in the latter part of this paper.

Ambient Networks Interfaces

Figure 1 also illustrates how the common, distributed control space encapsulates both legacy and future internetworking infrastructures (“Ambient Connectivity”) and provides generic interfaces (“hourglass” objects in Figure 1) that are independent of specific network architectures. Network entities interact with the new control space through these three interfaces.

Higher-layer applications and services use the *Ambient Service Interface* (ASI) to access a subset of the control space functionality. This subset includes functions like naming, location and context management, inter-domain management and traffic engineering. The ASI provides an API to Ambient Networks that is located between the control space and the applications at 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. The ASI also exposesi management capabilities and makes network context information available to the applications.

Connectivity resources interact with the control space through the *Ambient Resource Interface* (ARI), for example, to access multi-radio resource management, mobility and trigger processing. The ARI is located inside a node between the control space and the connectivity layer. It offers control mechanisms, which enable the control space to manage the resources residing in the connectivity plane. These resources can be routers, switches, radio equipment but also media transcoders, filters or proxies.

Finally, the *Ambient Network Interface* (ANI) facilitates communication between the control spaces of different networks, creating the shared, common control space that enables the advanced internetworking capabilities the Ambient Network project aims to achieve. 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 individual node of the network; the nodes collectively implement this core functionality as an aggregate.

Common Control Space Functionality

Some pieces of the control functionality shown in Figure 1 are required in the control space of each Ambient Network; they are what makes a network “ambient.” Figure 2 highlights these shared pieces of functionality. This common control functionality enables the plug-and-play interworking of the other control space functionality by providing a common architectural framework for intra-control-space communication, a control-space-wide resource registry and mechanisms for consistency management and conflict resolution.

One piece of this common functionality enables different functions within the ACS to communicate by exchanging messages with one another. Message-based communication among a set of participants requires some infrastructure and mechanisms. Participants need unique identifiers to enable unambiguous message delivery. A resolution mechanism must map these identifiers into locators for the specific message delivery mechanism. Furthermore, 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.

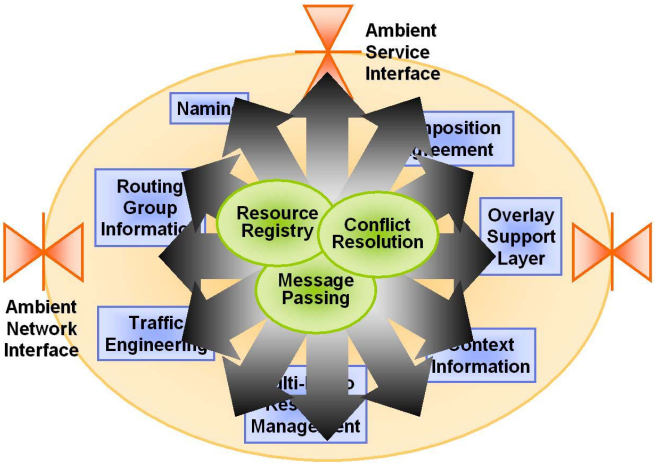

Figure 2. Common control space functionality.

i. Control Space Communication

These required features for message-based communication within the ACS are similar to what the Ambient Networks user plane abstraction offers (see below), but provide a consistent communication service on top of it. In some sense, the ACS can be seen as a distributed application or service implemented on top of the generic user plane.

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 connectivity 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.

Additionally, two communicating parties must agree on a specific encoding for the information they transfer. This capability is *not* part of the generic connectivity 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, are main requirements. Existing encoding schemes such as MIME [4], XML [3] or ASN.1 [5] may be readily adaptable for ACS communication.

Finally, the generic connectivity 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.

ii. *Resource 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. Note that although the registry is a logically centralized service, larger Ambient Networks will likely choose to implement it in a distributed fashion.

One purpose of the ACS registry is storage of information about connectivity plane entities such as network resources, services, specific hardware, links, sessions, policies and user information. 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 and consistency control functionalities 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 provides basic resource control functions to coordinate the different ACS components, for example, by providing access control. Although other components can provide access control for specific entities, the central repository requires a separate service that is independent from optional components. A second piece of functionality manages potential conflicts within the registry, e.g., if different entities aim to insert conflicting information.

iii. *Consistency Control*

Consistency control and conflict resolution mechanisms coordinate the concurrent, distributed decisions made by individual functions of the ACS. 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. A second group of inconsistencies can arise when a decision by a control function conflicts with the policies established for an Ambient Network by users, operators, or even automatically as part of a composition agreement.

Furthermore, 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 connectivity plane. The project is currently investigating an approach to do consistency control at different levels. Conflicts are first resolved locally within network elements; if this is not possible, resolution occurs within a functional area. Finally, at the level of the Ambient Control Space, generic conflict resolution functionality computes and maintains fairness relations between functions.

Connectivity Abstractions

Ambient Networks enable interoperation of legacy networks by providing a number of identical communication primitives independent of the underlying connectivity technology. Figure 3 shows a high-level overview of the two connectivity abstractions and how they integrate with the overall architecture. The *bearer* abstraction is exposed through the ASI to applications, while the *flow* abstraction enables the ACS to remain independent of the underlying network technologies.

Figure 4 gives a more detailed view of the connectivity abstractions. It shows four layers from the application at the top to the connectivity provided by a specific network technology at the bottom. All nodes are visible at the lowest layer, but higher-layer abstractions may eventually hide their presence from the communicating entities. Starting from the top, the connectivity abstractions are sessions, bearers and flows. Due to space limitations, this paper can only briefly discuss the specifics of these abstractions. A companion paper that is currently under preparation will discuss these topics in detail.

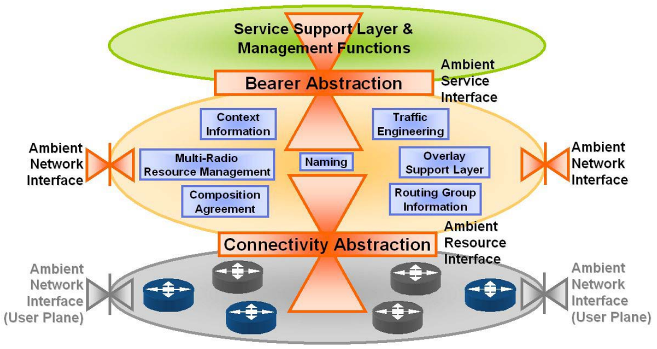

Figure 3. Connectivity abstraction overview.

Sessions are application-specific notions of connectivity: particular applications define the precise meaning in terms of mapping them to bearer(s). Bearers provide a connectivity abstraction for applications at the ASI. Flows are an abstraction of the basic connectivity provided by the underlying network technology at the ARI.

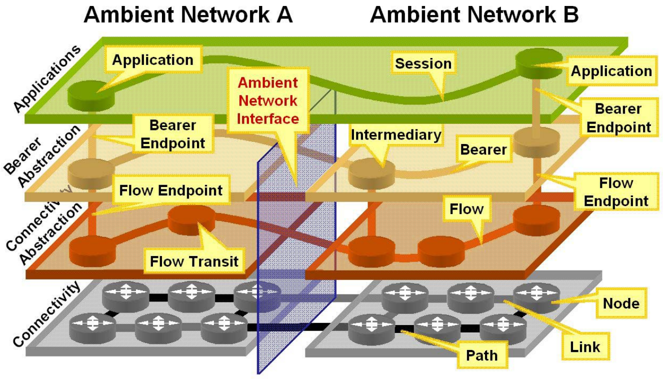

Figure 4. Detailed view of the connectivity abstractions.

A *flow* is an abstract view of the connectivity provided by the underlying network technology. 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 other types that is not necessary. A flow may pass through *intermediate resources* not explicitly tied to the flow but controlled through the ARI. The set of intermediaries 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. The flow transfers data between successive nodes using the underlying connectivity functionality. The control space may use the control and configuration capabilities of the ARI to request certain treatment of the flow by the connectivity layer. Flows transfer data transparently, with certain performance characteristics, which may include the level of integrity of the data. Mobility of a data transfer requires modifications to a flow because of modified locators or a new flow to be created and the old one deleted.

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 use 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 control the data manipulation functions of the control space. Certain applications, e.g., file transfers, can use simple bearers that require little other than what a flow provides. For other applications, e.g., voice, the bearer can become more complex and involve transcoding and media routing intermediaries. When an intermediary requires special treatment, the bearer maps to two flows – one from the source to the intermediary, and one from the intermediary to the destination. The indirection support of the naming framework described above controls this mapping of a bearer to a set of concatenated flows between the intermediaries. Multiple bearers may map to the same flow, i.e., flows can multiplex bearers between the same pair of locators.

Multi-Radio Architecture in Ambient Networks

This section gives an overview of the part of the Ambient Control Space that is closest to the radio interface: the multi-radio access (MRA). More thorough descriptions can be found in [6][7].

Plenty of prior work exists on multi-radio access topics, including IP mobility schemes for handover [8], joint radio resource management mechanisms [9], and different radio abstraction layers [10]. However, this work has only tackled partial issues, e.g., focusing on a limited number of specific RAs that could be tightly integrated, or proposing loose integration to handle many RAs but with limited support for joint resource management. The aim for the Ambient Networks MRA is an all-encompassing, flexible architecture considering all existing and future radio access technologies, and supporting different levels of coordination depending on operational modes and business relationships so that cooperation at the radio access level is possible even between competing actors.

The MRA architecture consists of two main components: Multi-Radio Resource Management (MRRM) for joint management of radio resources and load sharing between the different RAs; and Generic Link Layer (GLL), which provides a toolbox for unified link layer processing, offering a unified interface towards higher layers and an adaptation to the underlying radio access technologies.

A main feature of the MRA architecture is resource sharing and dynamic agreements between ANs, including different access providers, through composition. Other features are efficient advertising, discovery and selection of RAs, including the possibility for a user to simultaneously communicate over multiple RAs, in parallel or sequentially, and efficient link layer context transfers. Further, the MRA architecture supports multi-radio multi-hop communication using both moving and fixed relays. The MRA architecture uses the ACS infrastructure for communication between MRA components, consistent data storage by registries and conflict resolution.

MRA Architecture

A high-level view of the proposed MRA architecture is illustrated in Figure 5, showing functional blocks in a layered model, including user plane data flow and MRA signaling through the layers. Black arrows indicate control interfaces between different functional blocks, carrying information exchange and control commands e.g. for configuration or for measurement data retrieval. Note that only one communication peer (network or terminal) is depicted.

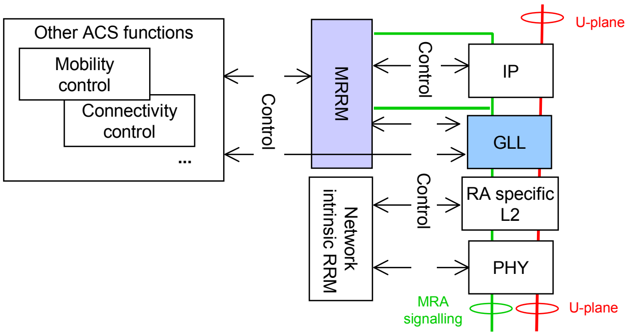

Figure 5: High-level MRA functional layer architecture.

The GLL is introduced on top of, and is partly replacing, the RA specific parts of L2. The toolbox of link layer functions within GLL provides a unified interface towards upper layers (IP and above) in the user plane and provides adaptation towards the underlying (remaining RAT specific) link layers.

The MRRM functions are built upon, or mapped onto the network intrinsic RRM functions, which belong to the underlying RA and are therefore not within the explicit scope of the AN MRA. The figure further illustrates the information exchange between MRRM, GLL and other ACS functions, here exemplified by mobility control and connectivity control.

The model suggests a functional split between MRRM and GLL. In general, GLL encompasses functions that are located close to the user plane of a data flow and/or need to operate on a relatively fine time scale. One example is selection of RAs for which a hierarchical distribution of functionality between MRRM and GLL is proposed, where GLL dynamically (fine time scale) handles the mapping of data flows to any of the RAs selected by MRRM (coarse time scale). Another example is that the GLL provides and reuses context information that is transferred between GLL entities at RA reselection for seamless access switching.

Multi-Radio Resource Management

MRRM performs spectrum and load management, and it coordinates decisions on different associated flows, where MRRM operations can be triggered either by system level operations or directly by flow level events, e.g., mobility. MRRM also establishes and maintains RAs that are possibly constituted of parallel multi-hop routes.

MRRM handles a number of RA coordination functions. *RA advertisements* inform about the presence, capabilities, and terms of use of a network. Proxy advertisements could be sent to advertise on behalf of other access providers or network nodes. *RA discovery* monitors candidate RAs and routes for specific flows. *RA selection* selects appropriate RAs for a given flow following an evaluation and an admission step. *Overall resource management* keeps global control of network resources through load sharing, excess QoS elimination, QoS downgrading, flow/session dropping, and dynamic spectrum control within or between RAs.

MRRM functionality may be distributed within and between ANs in various ways, which affects the type and amount of information that can be exchanged, and thus the possible degree of coordination. Centralized MRRM control is likely beneficial in any network of ANs belonging to the same administrative entity. Distributed MRRM control could be used in large networks for scalability reasons or when central control is not desired due to conflicting strategies (or lack of trust) between the involved entities.

Generic Link Layer

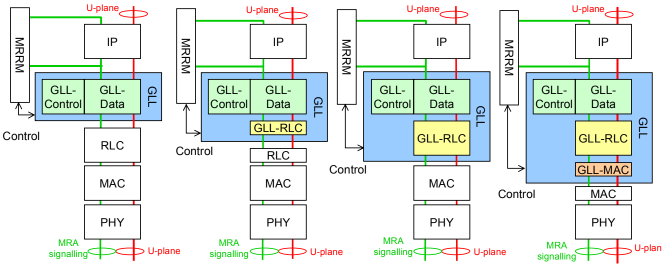

Figure 6: Four levels of GLL integration.

The GLL is defined as a toolbox of functionalities for unified link layer processing, which makes it possible to construct different alternatives of GLL integration. Figure 6 shows four different levels with, left to right, increasing degrees of integration corresponding to finer possible time scales of operation.

The model depicted in Figure 6 separates the GLL into the functional blocks GLL-C (control), GLL-D (data), GLL-RLC and GLL-MAC. GLL-C contains functions for the link layer configuration and the interaction with MRRM, including Access Selection Control, Resource Monitoring and Performance Monitoring functions. GLL supports mobility management functions and facilitates context transfer (GLL-D) and security management (GLL-D, GLL-RLC). Further, Access Scheduling (GLL-D, GLL-MAC), Buffer Management (GLL-D), Error and Flow Control, Segmentation and Reassembly (GLL-RLC) is supported.

The GLL toolbox of functionalities for unified link layer processing also supports two novel technologies: Multi-Radio Transmission Diversity for dynamic selection of multiple RAs (in sequence of in parallel) for transmission of a traffic flow, and Multi-Radio Multi-Hop for link layer support of multiple RAs over multi-hop routes. The latter includes mechanisms for multi-radio multi-hop ARQ.

Conclusion

This paper has presented an overview of the Ambient Networks architecture, focusing on several key functions of the control space. The control space provides common infrastructure for message passing, conflict handling, registry and connectivity abstractions. Furthermore, the paper presented a novel architecture for multi-radio resource management that is one component of this common control space. 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.

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.

References

[1] N. Niebert, A. Schieder, H. Abramowicz, G. Malmgren, J. Sachs, U. Horn, C. Prehofer and H. Karl. Ambient Networks - An Architecture for Communication Networks Beyond 3G. *IEEE Wireless Communications*, April 2004.

[2] N. Niebert, H. Flinck, R. Hancock, H. Karl and C. Prehofer. Ambient Networks - Research for Communication Networks Beyond 3G. Proc. *IST Mobile Summit*, June 2004

[4] D. D. Clark, J. Wroclawski, K. Sollins and R. Braden. Tussle in Cyberspace: Defining Tomorrow's Internet. Proc. *ACM SIGCOMM*, Pittsburgh, PA, USA, August 2002.

[3] Extensible Markup Language (XML) 1.0 (Third Edition.) *W3C Recommendation*. February 4, 2004.

[4] N. Freed and N. Borenstein. Multipurpose Internet Mail Extensions. *RFC 2045*. November 1996.

[5] Abstract Syntax Notation One (ASN.1) Specification of Basic Notation. *ITU-T Rec. X.680, ISO/IEC 8824-1:2002*, 2002.

[6] R. Agüero, E. Alexandri, J. Choque, K. Dimou, H.R. Karimi, G.P. Koudouridis, H. Lederer, J. Sachs, R. Sigle. Generic Link Layer for Multi-Radio Access Networks. Under submission *IST Mobile Summit 2005*.

[7] F. Berggren, I. Karla, R. Litjens, P. Magnusson, F. Meago, R. Veronesi, H. Tang. Multi-Radio Resource Management for Communication Networks Beyond 3G. Under submission *VTC Fall'05*.

[8] G. Wu, M. Mizuno, and P.J.M. Havinga. MIRAI Architecture for Heterogeneous Network. *IEEE Commun. Mag.*, pp. 126-134, Feb. 2002.

[9] J. Luo, R. Mukerjee, M. Dillinger, E. Mohyeldin, E. Schulz. Investigation of Radio Resource Scheduling in WLANs Coupled with 3G Cellular Network. *IEEE Commun. Mag.*, vol. 41, no. 6, pp. 108-115, 2003.

[10] P. Mähönen, N. Passas, G. Orphanos, L. Muñoz, A. Marshall, D. Melpignano, T. Inzerilli, F. Lucas, M. Vitiello, M. García, T. Saarinen. Platform-independent IP transmission over wireless networks: the WINE approach. *IEEE Pers. Commun.*, vol. 8, no. 6, pp. 32-40, 2001.