A Modular Access Gateway for Managing Intermittent Connectivity in Vehicular Communications

JÖRG OTT
Helsinki University of Technology, Networking Laboratory
PO Box 3000, FI-02015 TKK, Finland, voice +358-9-451-2460, fax +358-9-451-2474
jo@netlab.hut.fi
DIRK KUTSCHER
Technologiezentrum Informatik (TZI), Universität Bremen
Bibliothekstraße 1, D-28359 Bremen, Germany, voice +49-421-218-7595, fax +49-421-218-7000
dku@tzi.org

Abstract. The Drive-thru Internet architecture allows exploiting intermittent connectivity by temporarily connecting to IEEE 802.11 WLAN access points at the roadside from moving vehicles. This poses numerous challenges to a mobile user's equipment: extreme networking characteristics such as short periods of connectivity, unpredictable disconnection times, and vastly varying transmission characteristics. Heterogeneous WLAN hotspot installations may also require different authentication mechanisms and credentials. We have designed a mobile access gateway to deal with these issues on behalf of a user (group) in a moving vehicle and provide usable connectivity for applications without requiring manual operation. The gateway maximises the use of short connectivity periods by detecting network access providing signalling functions for local application processes. It also allows using dedicated radio equipment to prolong connectivity periods. Finally, in selected multi-user scenarios, further performance improvements are conceivable by sharing (non-confidential) information across users and applications.

1 INTRODUCTION

Mobile users and nomadic computing are presently supported by two classes of networks: Cellular networks aim at providing ubiquitous connectivity, even across different service providers. However, their price-performance ratio is rather poor and temporary disconnections may still occur for various reasons. IEEE 802.11 WLAN hotspots do not aim at seamless connectivity; their limited reach implies disconnection periods while the user is moving between locations. Hybrid approaches are pursued to keep users *always best connected* [1] [2] by combining access to different service providers or integrating wireless WAN and LAN to maximise connectivity, to improve the achievable data rate, and to minimise cost [3] [4].

In the Drive-thru Internet project, we rely on WLAN connectivity to provide affordable high-performance communications to mobile users and use dedicated as well as public hotspots for Internet access. The cost for establishing and operating WLAN hotspots can be quite low, and, since unlicensed operation is possible, deployment is not limited by regulations. As a result, WLAN has become an inexpensive commodity and the number of public hotspot installations is ever-increasing: besides hotels, cafés, etc., particularly airports, train stations, gas stations, and service areas are covered, i.e., places serving commuters and travellers on the road.[1]

Working with (public) WLAN hotspots usually requires manual user interaction, e.g., to (re)configure the WLAN interface, to authenticate with the wireless ISP [5] [6] [7], or to suspend, resume, and possibly reconfigure applications [8]. Working with hotspots from (potentially fast) moving vehicles means that only a short connectivity window is available for establishing network access and carrying out the actual communication tasks [7]. To enable hotspot usage and to allow such tasks spanning multiple hotspots without connectivity in-between, we have developed the Drive-thru architecture that conceals short-lived intermittent connectivity from applications [9] [10].

Regardless of the approach taken to provide wireless connectivity: all cases require sophisticated functions for network access, roaming, handover, authentication, cost and/or QoS optimisation, etc. Such functionality may be located in the end user's device (e.g., as offered by multi-access PC cards and associated software for laptops) or may be implemented in a dedicated access device, such as MAR [3], the mobile router in the eMotion project [11],

[1] Examples include Agip gas stations and MAXI service areas in Germany, Neste A24 gas stations in Estonia, Statoil in Norway, and Texaco service stations in the UK as well as truck stops in the US.

or the FleetNet access router [12].[2] These projects focus on offering ubiquitous connectivity (seamless handover and roaming) using mostly well-defined access control procedures and employing (variants of) mobile IP, i.e., they largely operate at the IP layer and below.

Performing access functions in separate devices offers numerous advantages: Dedicated radio equipment (including, e.g., external antennae mounted on top of a vehicle) provides better signal reception and prolonged connectivity periods [9]. Furthermore, a single router rather than multiple end systems accessing the same access point leads to more efficient utilisation of the wireless medium and thus better performance [13]. Finally, an access router may be augmented to perform higher layer functions such as TCP performance improvement or caching. The major disadvantages is that, with multi-user scenarios, the users share the radio resources (i.e., the capacity) available to a single mobile device. Thus, they may get a smaller individual traffic share each when contending with other devices. Also, the individual access charges across a common access router are difficult to account and bill for. Trust is considered less an issue since end users are expected to use secure protocols (at least for sensitive data) anyway.

This paper presents the design and implementation of a Drive-thru mobile access gateway (DT-MAG), a standalone device that serves mobile devices (within a vehicle) and connects them to hotspots along the road. In contrast to the aforementioned routers, the DT-MAG also performs transport and application layer functions that raise numerous issues regarding security and persistence of information. A flexible and modular design allows various scenarios to be accommodated simultaneously.

In section 2, we introduce the Drive-thru Internet architecture, in section 3, we review related work. We present usage scenarios in section 4 and derive requirements for the DT-MAG in section 5. The design and implementation of the DT-MAG are presented in section 6. Section 7 describes our measurement setup and reports on our results. In section 8, we review some key aspects of our modular design in a broader scope and conclude this paper in section 9 with a summary and a brief outline of future work.

2 DRIVE-THRU INTERNET

As mentioned above, the Drive-thru Internet project [9] aims at providing Internet services to mobile users moving at high speeds (e.g., in vehicles or trains). Access to Internet services is realised by exploiting conveniently located WLAN hotspots to which connectivity is temporarily established while a user traverses the hotspot's coverage area. The Drive-thru architecture allows existing and future applications to take advantage of such potentially short and unpredictable periods [10]. It relies on a connection splitting approach where a proxy in the fixed network maintains long-lived "end-to-end" connections on behalf of mobile clients that would otherwise be affected by intermittent connectivity [14]. Figure 1 depicts an overview of the Drive-thru Internet architecture.

The *Persistent Connection Management Protocol* (PCMP) is used for the communication between the mobile Drive-thru client and the Drive-thru proxy, allowing for creating and maintaining multiple persistent transport layer sessions despite frequent link layer disconnections. It is described in detail in [14]. PCMP uses regular TCP connections as underlying transport for a *PCMP connection*. Drive-thru client and proxy are authenticated upon PCMP connection setup in a connectivity island and a shared context is created that persists across disconnection periods. This shared context is used for subsequent management of application TCP connections. Following a split connection approach, they terminate/originate at the Drive-thru client and proxy and are multiplexed as otherwise independent *transport sessions* into the PCMP connection (with per session flow control): one application peer on the mobile node has a local (TCP) connection to the Drive-thru client which is forwarded through the PCMP connection to the Drive-thru proxy; the latter has a (TCP) connection to the corresponding application peer in the fixed Internet. The two connections between the application peers and the Drive-thru components are maintained independently of the connectivity between Drive-thru client and proxy.

Drive-thru Internet nodes must operate in today's *existing* WLAN infrastructure, which mostly consists of public hotspots. Hence, the mobile node must not just detect availability of a WLAN hotspot but also determine the hotspot operator and authenticate to gain network access via the hotspot. We have extensively discussed the requirements of automatic WLAN hotspot association, have devised an extensible component-based approach for heuristics-based automated authentication with different kinds of hotspots, and have shown how dedicated components can identify authentication mechanisms, perform policy-based authentication on behalf of the user, and notify PCMP and applications about (non-)availability of connectivity [7] [15].

Aggregating wireless communications from multiple endpoints, automatic hotspot association, and persistent connectivity management are examples of services that can be provided by a dedicated gateway for Drive-thru Internet environments on behalf of one or more end user devices. The Drive-thru gateway concept extends the ideas of mobile routers such as the Mobile Access Router (MAR) described in [3] or the mobile router in [11] by providing specific support for intermittent connectivity that can be used for various scenarios such as nomadic computing, WLAN mobility, and mobile networks. It is particularly the Drive-thru (session and) application layer functionality where a DT-MAG provides significant value-add beyond plain IP routing and mobility.

[2] Or the multi access router developed in the European 6WINIT IST project in support of IPv6-based mobile networks.

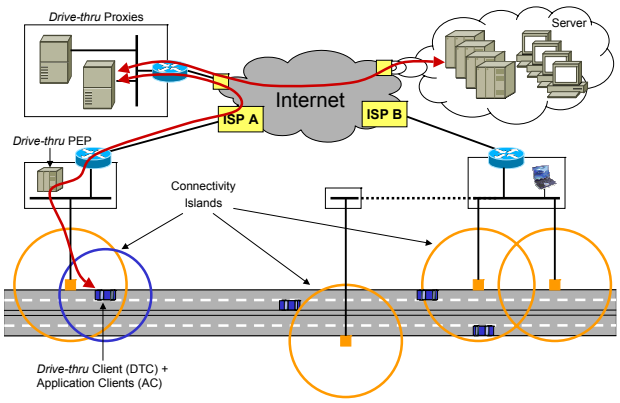

Figure 1: *Overview of the Drive-thru Internet Architecture*

Of course, intermittent connectivity cannot be completely shielded from the user and the user's applications. In some cases, this is not even desirable. In this paper, we also address possibilities to provide applications and higher layer protocols with hints about the current connectivity status, e.g., to adapt protocol and application behaviour to the varying network conditions. For example, to make use of short and unpredictable connectivity phases efficiently, it is useful to have the "connectivity managing function" notify higher layer instances such as the PCMP layer when connectivity has been established (or lost). This can avoid costly probing processes and can thus help to improve the overall performance of the system. This is especially important because employing a mobile access router—instead of connecting to the access network directly—separates the user device from the network link interface, making it more difficult to assess the network status directly.

3 Related Work

Our work from the Drive-thru Internet project described in this paper addresses vehicular network access in general and focuses on dedicated gateway devices and the implications for the mobile users' devices. IP communications on the road has independently been studied in FleetNet [12] and Networks on Wheels, with a different focus and slightly different goals though: both projects primarily target inter-vehicle communications in wireless ad-hoc networks for traffic-related control information and data sharing across vehicles, the latter of which is also addressed in Hocman [16]. Hybrid network access for vehicles has been addressed, e.g., in OverDRIVE[3] and IPonAir [4].

The multi-homed mobile access router (MAR) for moving vehicles [3] dynamically instantiates new channels, provides bandwidth aggregation, and dynamically shifts the load according to channel characteristics and bandwidth demand. MAR uses multiple wireless channels simultaneously to exploit *network diversity* in different dimensions, i.e., with respect to radio technology, with respect to operated networks relying on the same technology, and with respect to different channels for a given technology. The objective is to provide a more powerful and more robust connectivity, where disruptions of one access link do not necessarily lead to a complete loss of connectivity. While, according to [3], first experiments have shown that there is typically a substantial overlap in terms of coverage that can be exploited to reduce the number of disruptions and to increase the overall throughput, disruptions can, however, not be completely avoided.

Several dedicated access gateways or mobile routers have been developed. For example, in [17], a CDMA2000-1xEV-DO/WLAN gateway is proposed that can be operated in different network configurations and has the ability to move a user's CDMA session from a user terminal to the gateway system and to resume the session over the local WLAN interface. In July 2005, NEC Corporation has presented the *Litebird* prototype, a mobile router with two access link interfaces (3G and WLAN), capable of utilising both for improving the seamlessness in highly mobile scenarios. Based on commodity Linux-based WLAN access point and router platforms, many research and commercial projects[4] have started to develop application-specific support functions such as VPN gateways, IPv6 tunnel endpoints, and VoIP application gateways.[5]

The IETF NEMO (network mobility) effort is concerned with managing the mobility of an entire network that changes its point of attachment and its IP reachability. The NEMO Basic Support model [18] is an extension of Mobile IPv6 and relies on a bidirectional tunnel that is established between a home agent and the mobile router so that mobility is transparent to the nodes in the mobile network. The NEMO approach itself does not consider disruptions caused by loss of connectivity or hand-over delays. Similar to Mobile IP, it is a layer 3 solution that addresses reachability when changing the point of attachment. In order to enhance the usability of network mobility, e.g., in moving-vehicles scenarios, the mobile router could provide multiple interfaces, i.e., be multi-homed. Work in progress in [19] analyses how NEMO can accommodate multi-homing goals such as permanent access, redundancy, load sharing, load balancing, and user preference consideration. The InternetCAR project [20] [21] is exploring the possibility to enable Intelligent Transport Systems (ITS) by installing NEMO based routers in cars, however without addressing disruption tolerance explicitly.[6]

While NEMO based approaches basically provide IP layer mobility, thus maintaining IP end-to-end communica-

[3] http://www.ist-overdrive.org/

[4] In September 2005, Vodafone has announced a UMTS/WLAN gateway based on the Linksys WRT54G access point.

[5] Many of these projects are leveraging the *OpenWrt* Linux distribution for the Linksys WRT54G WLAN access point.

[6] Network mobility has also been addressed by the European IST projects DAIDALOS and Ambient Networks.

tion semantics, there are other approaches for mobile Internet access that explicitly address intermittent connectivity and offer indirect access to information, e.g., by using content caching and prefetching. The Boeing Connexion system[7] is an in-flight entertainment system that provides Internet access to passengers in aircrafts via an aircraft router that communicates with ground stations via satellites. The on-board network is based on Ethernet and WLAN. In addition to on-board content, e.g., movies, Connexion provides a direct Internet connection at selected times during a flight. While the goal is to provide permanent connectivity, disruptions cannot be completely ruled out, and the system is not available during take-off and landing. Deutsche Bahn, the major German train company, has started to offer WLAN-based Internet access in trains which is realised by WLAN-UMTS gateways on the train.[8] On train lines where permanent connectivity cannot be achieved, the system offers offline access to prefetched content (web resources) updated at train station stops—where the on-board router uses WLAN hotspots in train stations.

Dealing with temporary connectivity loss is studied in the eMotion project [11]. It implements a dedicated mobile router to provide wireless network access, addressing multi-provider support at the IP layer and temporary connectivity interruptions at the transport layer, respectively. Finally, delay-tolerant networking (DTN)[9] [22] deals with intermittent connectivity for asynchronous applications and Drive-thru clients and proxies conceptually resemble DTN routers to a certain degree. Many of the aforementioned projects also address authentication but all of them assume a well-known authentication mechanism. Gaining knowledge about heterogeneous hotspots and the corresponding WISPs may be achieved by means of—standardised—service discovery mechanisms as has been discussed, e.g., for Wi-Fi alliance's smart client authentication [5] and FleetNet [23].

4 Mobile Usage Scenarios

Mobile access routers in general and the DT-MAG in particular can be used in different settings all of which feature mobile users with laptops running their respective application clients (ACs in the figure below). We can differentiate at least according to the following aspects: single-user vs. multi-user; dedicated gateway system vs. complementary software component on a user's laptop; and according to the accounting/trust relationships between users and the DT-MAG in the multi-user case. This results in the following scenarios:

1) A mobile user without a vehicle (or without any supportive infrastructure within one) uses arbitrary hotspots for her communication needs. Obviously, such a user needs to carry her client-side Drive-thru infrastructure with her at all times (e.g., on her laptop). Using only her laptop for all functions, she is responsible for herself and there is no need for shared accounting/trust. Note that this case also includes multiple independent users in the same vehicle.

2) For a user alone in his car, a similar 1:1 trust relationship exists. However, wireless access, persistent connections, and application support may be provided by a physical DT-MAG device that is part of the car's communication infrastructure.

3) Multiple users in a vehicle may use the same DT-MAG components with a shared trust relationship, e.g., for a family travelling together in a car.

4) Multiple users in a vehicle may use the same DT-MAG components without a shared trust relationship, e.g., passengers in a bus or on a train.

The degree of support provided by (or requested from) the vehicular infrastructure may differ (figure 2): as a simple access router, a DT-MAG may just provide wireless connectivity (the access function, AF, usually including authentication) or may also implement a shared Drive-thru client (DTC) offering persistent connections and application support. In the latter case, further application-specific functions (e.g. a shared web cache) may also be realised on the DT-MAG.

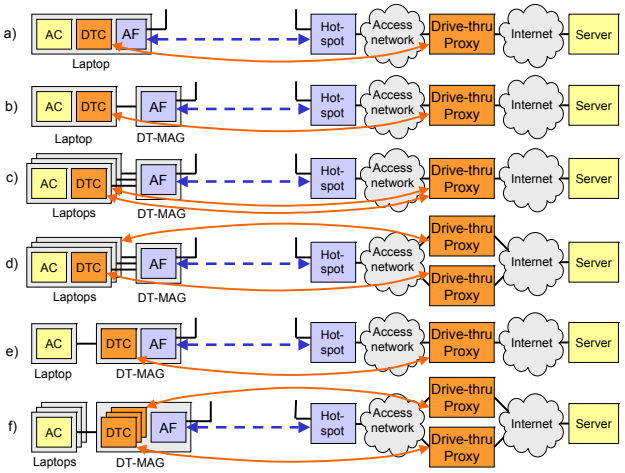

Figure 2: Drive-thru Mobile Access Gateway usage scenarios

Except for case 1—where the user is in full control of her entire Drive-thru infrastructure at all times (fig. 2a)—we can further differentiate whether *i)* application sessions may persist across a user entering and leaving a vehicle or *ii)* they only last while the vehicular infrastructure components are available. Some users travelling with their laptops will presumably prefer type *i)* as this allows them to maintain persistent application sessions at all times (particularly if their time aboard the same vehicle is limited; e.g., short-distance commuters). This implies that the DT-MAG support is limited to wireless access functions (fig.

[7] http://www.boeing.com/connexion/
[8] http://www.imice.de/
[9] http://www.dtnrg.org/

2b–2d). In contrast, users with their own laptops who remain aboard a vehicle for an extended period of time (such as long-distance travellers) and users accessing the Internet via a vehicle's built-in devices will find type *ii)* sufficient. In this case, the DT-MAG may provide persistent connections for the applications (fig. 2e and 2f).

Finally, for hotspot access and Drive-thru proxy communications, mobile users need to authenticate with the respective service provider: If a DT-MAG provides wireless access (2b–2f), the users are required to share the access provider (since accessing multiple WISPs from the same wireless station adaptor is usually not supported in practise). If the DT-MAG also acts as a Drive-thru client (2e–2f), the Drive-thru proxy (and hence the Drive-thru provider) must also be shared by the users (2e), unless the Drive-thru client can interact with multiple Drive-thru proxies and offers users a choice (2f).

5 Mobile Access Router Requirements

In a Drive-thru environment, a mobile node must find and gain access to hotspots quickly to make most use of the potentially short connectivity period. While portable computers may well perform this function on their own, a dedicated DT-MAG device for Drive-thru clients may benefit users in numerous ways: it may offer dedicated high performance radio equipment, may be specifically designed to deal with rapidly changing connectivity, and, if shared among multiple users, may improve network utilisation. Measurements have shown dramatic performance differences when comparing Drive-thru WLAN usage with and without external antenna [9].

The following list outlines the requirements for a DT-MAG, motivated by the goal to move as much Drive-thru-specific functionality as possible into the DT-MAG device to minimise dependencies on user equipment. The above scenarios are used to infer requirements for the distribution of these functions between a DT-MAG built into the vehicle and the user devices.

A. Detecting network access. The detection of network access involves both sensing link layer connectivity and testing whether a handover has been performed, i.e., in order to determine that a new IP stack configuration process has to be initiated. Detecting (and establishing) link layer connectivity is a perfect fit for a DT-MAG device since this function does not involve user-specific action and hence is easily sharable—and all users may benefit from router support for additional link layer technologies. E.g., in hybrid networking environments [1], the task for simultaneously probing different network interfaces and establishing network access on the optimal interface can best be provided by a dedicated device.

B. IP auto-configuration. When link layer connectivity has been established, the IP auto-configuration process must be initiated, typically DHCP for IPv4 in today's hotspots, but other configuration mechanisms (if any) must also be detected automatically. The DT-MAG is a good match for this function if only a single IP address for the entire vehicle is available per wireless network that has to be shared by all users in a network address translation fashion. Similarly, a DT-MAG needs to provide plain IP access router functionality to the user devices (e.g., DHCP, NAT, and IP routing) to allow them to also connect transparently to the Internet when connectivity is available.

C. Network selection, authentication, and accounting. Typical hotspot installations rely on web-based access methods, where users have to authenticate themselves before obtaining Internet access. The Wi-Fi Alliance has standardised the *Universal Access Method (UAM)* [5] that specifies, among other operational details, how HTTP requests from the mobile user are to be intercepted and redirected via a TLS connection to a provider's web server as well as how the web pages including the HTML forms for login/authentication and logout are to be structured. Larger WLAN hotspots may support multiple operators sharing the radio infrastructure and thus require a network selection step before the actual authentication (which is not defined in UAM). User (or vehicle operator) policies should be able to govern which service provider(s) are used [15].

While such web-based access authentication methods are generally intended for human users, it is an obvious requirement for the Drive-thru environment that the authentication process be performed automatically [7]. Furthermore, numerous different variants need to be taken into account because web-based authentication portals are rarely UAM compliant but rather exhibit provider-specific properties. To allow for encompassing use of WLANs, various other (emerging) authentication schemes have to be supported as well as discussed in [15].

As usually no hints are available that would allow identifying the respective authentication scheme, let alone the service provider or other access-related information, the mobile device has to revert to a heuristics-based trial-and-error approach. With all users on the vehicle sharing the same network access, this fairly complex task may also be taken up by the DT-MAG.

Finally, different authentication schemes operate at different layers: open, WEP-based, and 802.11i authentication at the link layer, IPsec and other VPN-style authentication at the IP layer, and web-based mechanisms (that are prevalent for public hotspots) at the transport and application layer [15]. As a result, there is no clear sequence of actions: depending on the authentication scheme, IP configuration may be carried out prior to (IP and higher layer) or after (link layer) authentication. With VPN approaches, IP autoconfiguration may even have to be carried out repeatedly. And, in static cases, IP autoconfiguration does not occur at all. Any split of responsibilities between a dedicated access gateway and one or more user devices has to take these alternatives into account.

D. Service detection. Mobile nodes may benefit from hotspot-local service announcements indicating available WISPs, authentication methods, and tariffs to avoid sophisticated heuristics and trial and error methods for authentication. Furthermore, hotspots could announce additional Drive-thru or other services [7]. Service announcements need to be interpreted by the DT-MAG and be used in conjunction with e.g., authentication. As they may also be useful for client applications running on the user devices, the DT-MAG must also be capable of distributing these announcements to clients on the local network.

E. PCMP client functions. The PCMP client is responsible for initiating a PCMP connection to the corresponding Drive-thru proxy (including user authentication) and for resuming application sessions when a connectivity island becomes available—as well as to suspend these sessions and tear down the PCMP connection when connectivity is lost. This particularly incurs maintaining the state necessary for persistent application sessions. Hence, using PCMP client functions on the DT-MAG is only feasible if the application sessions need not persist longer than the user is aboard the respective vehicle. Otherwise, the PCMP client needs to reside on the user device.

As the PCMP client also authenticates with the Drive-thru proxy, this requires either a shared account (and hence trust) or some accounting relationship between all users. Alternatively, the PCMP client may establish per-user PCMP connections—which, however, would require means to securely delegate user authentication to the DT-MAG. For simplicity, we will restrict our further considerations to the former, shared account case.

F. Application-specific functions. Access routers in aircrafts, trains, etc. often provide functions to improve application performance, e.g., web caches, SMTP proxies, etc. Such functions are also applicable to the Drive-thru environment. However, to realise them on the shared DT-MAG, it needs to have access to the application connections and hence must run the—shared—PCMP client to terminate the PCMP sessions. Sharing the PCMP clients implies, however, that application data of different users may be stored temporarily in the DT-MAG: while sharing public content (such as web pages), e.g., for caching is less an issue, entrusting a shared PCMP client with personal data (such as emails) is rather problematic. This calls for a DT-MAG approach that allows different (user-controlled) choices for those applications that could benefit from sharing and those that do not.

G. Triggering applications. All functions that are provided on the DT-MAG must be able to notify the user's applications about state changes that may be of interest to them (e.g., when connectivity becomes available or is lost, PCMP authentication completed, etc.). This allows applications to react quickly to such changes and particularly to make efficient use of short connectivity windows. [10]

[10] Conversely, the user applications (in our case PCMP) might indicate whether there is a need for connectivity in the first place: each application might emit *data-to-transmit* or similar events based upon which the decision could be taken when to establish Internet access (and when to tear it down again) so that unnecessary load in the hotspots could be avoided. In addition, applications could also provides hints to the PCMP layer when they experience connectivity problems, thus allowing for a faster recovery after disconnection. These functions are subject to further study.

Further requirements may arise depending on how the access to the DT-MAG from mobile users within a vehicle is organised: e.g.,, if WLAN is used aboard an aircraft or in German rail's ICE trains, an on-board access control system similar to UAM may need to be deployed in addition. As such mechanisms are orthogonal to the Drive-thru functions of the DT-MAG, they are not considered further in this paper.

6 DT-MAG Design

To support the different potential usage scenarios and their specific requirements, we have designed a modular architecture for the DT-MAG that allows to assign the individual components to different devices (DT-MAG or user device) as depicted in figure 3. We have largely mapped the functional requirements identified in the previous section to individual software modules. The *ConnectivityDetector* is responsible for monitoring one or more (wireless) links and providing information about available networks and also performs IP layer autoconfiguration. It always runs on the DT-MAG. The *AutoAuthenticator (AA)* authenticates the DT-MAG with the WISP—and might also listen to potential service announcements from the hotspot provider on a well-defined multicast transport address. The AA may run on the DT-MAG and on the mobile nodes—in the latter case, however, the AA will automatically be disabled when the mobile device notices that a DT-MAG already provides the corresponding functionality. Finally, the PCMP client provides persistent transport connections across connectivity islands. If running on the DT-MAG, additional application functions such as caching may be integrated; otherwise, only the application-specific adaptation modules may be included.

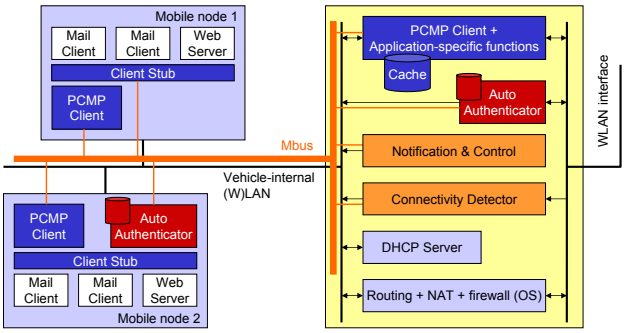

Figure 3: Structure of DT-MAG and associated mobile nodes

These entities interface to each other using a message-oriented group communication mechanism for coordination in component-based systems—the *Message Bus (Mbus)* [24]. Mbus is a platform- and programming-language-independent coordination protocol for application components located on a single host or within a local network, thus enabling the flexible distribution of gateway components depending on the specific usage scenario.

Mbus provides automatic location of application modules, aliveness detection and flexible group and point-to-point communication mechanisms. The Mbus framework is typically applied to specific application scenarios by defining classes of Mbus entities (the application components), corresponding addressing schemes, and a set of application-specific messages. The Mbus communication for interface management is depicted in figure 4. We have defined Mbus messages for the following basic trigger messages all of which convey soft-state updates:

1. Indication of availability (and loss) of network connectivity including link layer type, network name (e.g., SSID), signal strength (SNR), and L2 transmit rate. This function is provided by the `SCAN_RESULT` messages in figure 4.

2. Completion of IP layer autoconfiguration including external stack configuration parameters (IP address, netmask, DNS server, etc.). This function is provided by the `CONNECT` message in figure 4.

3. Information about the necessary authentication procedures (at the link or IP layer) and whether this authentication needs to be performed by the individual mobile nodes or whether this is taken care of by the DT-MAG. This function is provided by the `SCAN_RESULT` messages in figure 4.

4. An indication that authentication has completed and that the access network is now ready to use. This function is provided by the `STATUS` message in figure 4.

The Mbus-based communication can be separated in *access router-internal* and *external* communication. The internal communication is used for coordinating the internal gateway modules that are required for establishing Internet connectivity, and the external communication is used for signalling the current connectivity state to external components such as user PCMP modules and applications. The internal status notifications are aggregated and evaluated by a dedicated module and can trigger sending external status notifications. The WLAN network access function is implemented by three modules: the *ConnectivityDetector*, the *AutoAuthenticator*, and the *notification and control module*.

The *ConnectivityDetector* is monitoring the WLAN interface and performs the actual WLAN interface configuration and the WLAN association procedure. It generates two Mbus messages: `if.status` and `scan.result`. The `if.status` message is sent periodically for reporting the status of the WLAN interface and the current WLAN association, e.g., it is used to report the signal quality and the MAC address of the currently associated access point. `scan.result` is used to report the result of WLAN scanning processes. The message contains a list of available access points including specific information such as 802.11 channel, noise and signal level, ESSID, MAC address, and security information (e.g., WPA, WEP, open access).

The *AutoAuthenticator* manages the WLAN connectivity based on status events received from the interface manager. It monitors the connection status and decides, based on the user provided configuration, when to associate with a new WLAN access point. When the access module decides to associate with a new access point, it sends the `if.connect` message to the *ConnectivityDetector* that initiates the association process. This message includes parameters such as the access method (WEP, open access) and, if required, the WEP key to be used).

The *notification and control module* monitors the current state of the *ConnectivityDetector* and the *AutoAuthenticator*, and aggregates this information for external clients such as a PCMP module. It multicasts periodic `status` messages that describe the current interface status as one of `disconnected`, `disconnecting`, `connecting`, and `connected`.

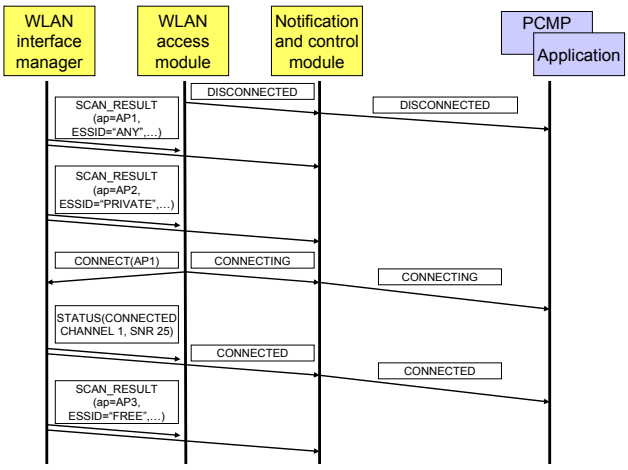

Figure 4: *Local Mbus communication for interface management*

A sample Mbus coordination session for interface management is depicted in figure 4[11]. The *AutoAuthenticator* coordinates the network association and notifies the notification and control module that represents the interface to external modules. In this sample message flow, the *ConnectivityDetector* reports two available WLAN access points from two different domains. We assume that the second access point is not available for public use and that the

[11]The message names are abbreviated for readability.

access module has no configuration that would allow using this access point. Therefore, the first access point is selected. In this example, the association has been successful, and the *ConnectivityDetector* starts to periodically report the status of this association. While being associated to this access point, the *ConnectivityDetector* may detect other available access points and will report these to the access module. The access module can use this information, together with the current link status and its configuration data for deciding when to perform a handover (not shown in figure 4).

On the mobile node, we provide a simple configurable TCP relay that is used as an outbound proxy by each application to be enhanced by Drive-thru services. It allows standard applications (such as mail client and web browser) to be configured with a static peer/proxy address—i.e., an address on the local machine—and redirects incoming connections: to a local PCMP client on the mobile node or to the PCMP client in the vehicle as determined by means of the local announcements.[12]

7 Testing and Evaluation

Our initial DT-MAG prototype implementation runs on a Linux-based laptop with a WLAN interface featuring an external antenna and an Ethernet interface for local in-vehicle communications (later referred to as the *laptop-only* configuration). The ConnectivityDetector, the AutoAuthenticator, and all related modules (such as the DHCP client) run on the DT-MAG. Finally, the PCMP client may be run either on the DT-MAG or on a user's mobile node, depending on the usage scenario. The user applications run on the client device presently connected via Ethernet but may also run on the gateway, e.g., in case of built-in computers in a vehicle, or may be distributed across both. While the present version of the integrated PCMP client provides just persistent transport connections, a stand-alone implementation already supports POP3 and SMTP as application protocols as noted above.

In addition, we have developed a special variant of the DT-MAG system that runs on an embedded device that is closer to a dedicated "car router". We have ported the DT-MAG software to the Linux-based Linksys WRT54GS WLAN access point with the following system specification: IEEE 802.11b/g WLAN interface, 4-port Ethernet switch, 200 MHz MIPS CPU, 32 MB RAM, 8 MB Flash. We have used the *openWrt* Linux distribution[13]. In this case, the WRT54GS acts as a router for a mobile network in a vehicle. For our measurements, the WRT54GS was connected to the laptop (the user device) via Ethernet, using one of the built-in Ethernet ports.

[12] A future optimisation could be to transparently capture the application's TCP packets and terminate the respective connection without any manual reconfiguration required for the application.

[13] http://www.openwrt.org/

Due to memory constraints, we were not able to directly port the full DT-MAG version. However, the modular design allowed us to split the functionality and run the *ConnectivityDetector* on the WRT54GS while moving the *AutoAuthenticator* to the laptop (the user device). The underlying Mbus coordination mechanism makes this split transparent to the software modules, so that the *AutoAuthenticator* obtains the link interface status over the local network.

Figure 5: *DT-MAG components used in measurements*

Figure 5 depicts the DT-MAG components we have used for our measurements: For the embedded system, the Linksys access point is connected to an external antenna and talks to the laptop via full duplex 100Base-T Ethernet. For the laptop-only configuration, the laptop-based prototype uses a Buffalo Airport eXtreme WLAN card that connects to the same external antenna. In both settings, the laptop is also connected via USB to an external GPS receiver for correlating measurement values to car positions and for recording the current speed.

We have carried out numerous measurements with the laptop-only and the WRT54GS-based system to validate the feasibility of our approach and to assess the performance of both configurations with respect to WLAN characteristics, connectivity establishment times (WLAN detection, DHCP-based client configuration), hotspot association (UAM login times), and TCP throughput.

We have chosen three different settings: in the laboratory (for reference), on the Autobahn, and in an urban environment. In all three scenarios, we have emulated a public WLAN hotspot by connecting an IEEE 802.11g access point to an authentication server (a laptop running *NoCat*[14]). For the lab setup, we have put the access points and the WLAN stations in a single room at Universität Bremen and measured the behaviour without mobility. For the Au-

[14] http://nocat.net/

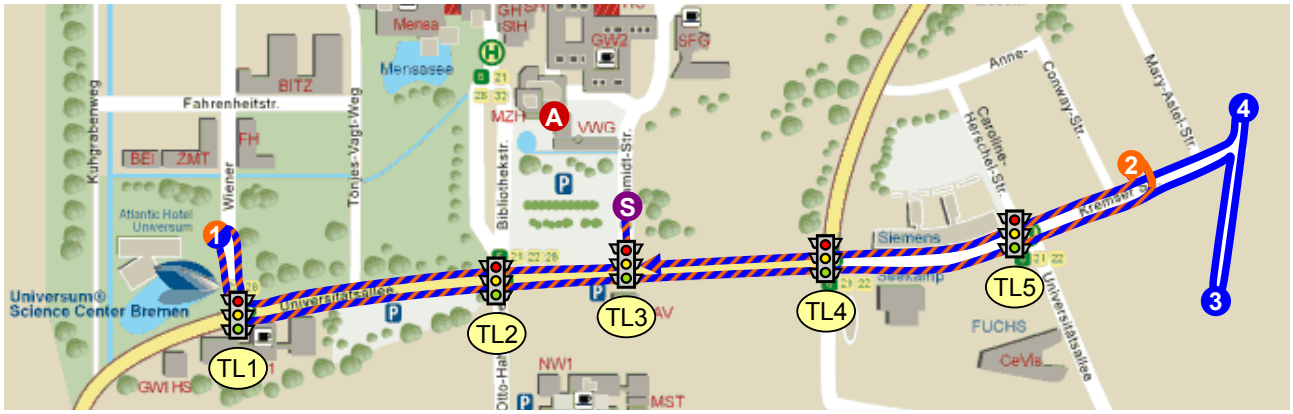

Figure 6: *Urban measurement courses*

tobahn setup, the access point and antenna were mounted on a pole next to the road in our "standard" rest area on the A27 between Uthlede and Hagen (at N 53° 20.484, E 8° 36.425), an otherwise rather isolated area.

For the urban setting, depicted in figure 6, we mounted the access point and the antenna on the fifth floor inside an office building on the campus of Bremen university behind a window (near N 53° 06.3843, E 8° 51.1349, (A) in the figure) facing the street Universitätsallee (which we used for the measurement drives). We performed system configuration at point (S). For the laptop-only measurements (orange lines), we drove back and forth between turning points (1) and (2)—which are chosen so that they are out of the radio range of the AP. For the Linksys measurements (blue lines), we needed to extend the course due to the (perceived) farther reach of the WLAN hardware and chose turning points (1) on one end and (3) and (4) on the other. Due to its urban nature, many other WLAN access points with different SSIDs and authentication schemes were visible including the university's campus WLAN.[15]

In both mobile settings, we repeatedly drove past the WLAN access points, turned and passed again. In contrast to the autobahn setting where we could maintain a constant velocity of 120 km/h, traffic conditions and the course itself in the urban setting led to the velocity varying between 0 km/h (at red traffic lights) and some 50 km/h. The traffic light are marked as TL1, TL2, ..., TL5 in figure 6.

For all tests, we measured the net TCP throughput for data transmission from a laptop in the fixed network to the mobile node, with the fixed node sending at maximum rate. We used our self-developed measurement infrastructure, including a TCP traffic generator called *tcpx* and various analysis scripts [14]. Each test run consisted of a single *tcpx* session via PCMP that was maintained across several passes through the artificial WLAN hotspot. The PCMP client was informed via the Mbus about the connection to the wireless LAN *after* successful authentication and so could initiate/resume data transmission in a timely fashion.

Our measurements provide quantitative information about connectivity periods, the delays between radio link detection and its use for application data, and the TCP transmission characteristics. However, we have conducted only a few proof-of-concept measurements on the autobahn and obtained results only for the laptop-based DT-MAG.[16] Instead, we have focused on urban measurements that are environmentally more challenging as discussed below. The results we have obtained from these measurements (some 15 for the laptop-only and some 20 for the Linksys setup) are, however, only of limited statistical significance for at least two reasons: 1) the changing environmental conditions (such as velocity and other vehicles on the road) and 2) the unpredictable interference with other WLAN networks (which are important to test with, however).

Nevertheless, the sample data presented below—a set of successive measurement runs representatively chosen from a larger collection of measurements to show important observations—is roughly repeatable. While further measurement results are clearly desirable for validation, our findings so far are sufficient to prove the feasibility of our implementation, help to identify some trends and issues, and also point to potential for future optimisations. Note that, except for the achieved throughput (which is due to a different hardware setup, active scanning, and, for the urban scenario, also due to interference), these results match fairly well our previous measurement runs carried out for different purposes [10] [14] [7].

[15] During our measurement runs, we found 29 access points using 16 different SSIDs with the laptop-only setting and, (partly) due to the slightly longer course, 47 access points using 30 different SSIDs. with the Linksys system.

[16] It turned out that the performance of the embedded DT-MAG implementation on the Linksys access point degraded significantly when moving higher speeds, e.g., at 120 km/h, our Autobahn reference speed: the WLAN interface detection mechanisms that worked reliably in the lab did not operate properly at higher velocities. We have analysed this behaviour in the lab and will perform further measurements with the embedded DT-MAG implementation in the future.

Table 1 gives an overview of one of our test runs on the Autobahn and in the urban setting, listing the time required for DHCP-based autoconfiguration, for UAM-based authentication (if any), and the total access delay as well as the usage period denoting the time the link was used for actual *tcpx* data transfer via PCMP. The measurements *Autobahn 1* and *2* were performed with plain WEP access control[17] (configured based upon the observed SSID), *Autobahn 3* and *4* as well as all *Urban* measurements had UAM enabled. The long UAM authentication delay in *Autobahn 3* is caused by dynamically loading the required modules upon first contact. The total connectivity period (during which the AP is usable for the mobile node) is comparable to previous experiments [10] [14]. *Urban nb* indicates notebook-only setups and *Urban ap* those using the Linksys access point. The Ids are used for reference and *a1–a4* and *b1–b4* also correspond to the plots in figure 8. The urban measurements have been performed driving in an alternating fashion in both directions: for measurement *a1* we drove from reference point 1 to reference point 2, back for *a2*, and so on. For measurement *b1* we again drove from point 1 to 4. As noted above, the traffic conditions and interference in the urban scenarios were highly variable and thus caused the significant differences in connectivity periods.

Location	Id	DHCP Delay	Auth Delay	Total Delay	Usage Period
Autobahn	1	1.91 s	–	1.91 s	55 s
Autobahn	2	1.52 s	–	1.52 s	76 s
Autobahn	3	4.87 s	25.00 s	29.87 s	26 s
Autobahn	4	2.83 s	10.00 s	12.83 s	54 s
Urban nb	a1	6.38 s	3.00 s	9.38 s	70 s
Urban nb	a2	16.31 s	2.00 s	18.31 s	118 s
Urban nb	a3	51.15 s	2.00 s	53.15 s	5 s
Urban nb	a4	10.94 s	2.00 s	12.94 s	46 s
Urban ap	b1	5.56 s	2.02 s	7.58 s	66 s
Urban ap	b2	40.91 s	2.02 s	42.93 s	57 s
Urban ap	b3	10.69 s	2.02 s	12.71 s	55 s
Urban ap	b4	45.05 s	5.04 s	50.09 s	106 s

Table 1: *Access delay and connectivity periods when connecting to a WLAN access point via the DT-MAG*

From table 1, we observe that the access delay incurred is ideally not larger than 20 s including link detection, DHCP-based auto-configuration, and UAM. The major variability in the urban measurements arises from DHCP which takes typically much longer than for the autobahn measurements.[18] This effect can be explained by the varying traffic conditions in conjunction with an implementation and reporting characteristic of the DT-MAG: it notices WLAN access points from beacons well before the signal is strong enough for successful association. As the *ConnectivityDetector* does presently not distinguish a separate "associated" state, it reports a WLAN immediately upon beacon reception. If the car moves constantly, association follows right away and the initiated DHCP procedure will succeed (which is always the case on the autobahn but only sometimes in the urban scenario). In the urban measurements, it frequently occurred that the WLAN access point has been discovered when driving from reference point 1 eastwards, however, the link only became usable after passing traffic light 2. When there was slow traffic or even a red light at *TL2*, this phase could last 20 seconds or more during which the DHCP procedure does not succeed (see the discussion below).[19] It should be noted that, despite the unreliable broadcast transmission of DHCP requests and responses without 802.11 error control, analysing the tcpdump traces revealed only rare DHCP packet losses (adding only a second or two further delay).

Figure 7 shows the activities of the DT-MAG and the net application data rate during *Autobahn 4*. As can clearly be seen, the access delay of some 13 seconds only occupies the entry phase (entering the hotspot [9]), leaving most of the production phase (the phase with good connectivity) for the actual data exchange. The figure also depicts when the *ConnectivityDetector* determines the link loss from monitoring the wireless LAN card shows the delay before notifying the application (PCMP) about this event of some 15 seconds. The artificial delay avoids frequent suspend/resume operations of the application in case the connectivity is regained shortly afterwards and thus prevents extra overhead and oscillation. This comes at the penalty of increased switchover latency if different access points are close by, thus potentially leading to wasted communication opportunities.[20] Therefore, we consider adapting this notification delay dynamically depending on the density of available access points in the surrounding area.

Figure 8 shows the SNR and net application data rate for *Urban a1–a4* and *b1–b4*. We observe that the data rate is lower than for the Autobahn scenario which must be attributed to significantly lower SNR due to the positioning of the access point and also to more interference (from multipath effects, from numerous other wireless LANs) and attenuation (buildings, trees, other cars)—which is supported by the significant variation in maximum throughput.

Autoconfiguration and automated WLAN access take place between the respective first indication of a radio signal and the start of the data transmission. This period that can be rather short as, e.g., *a4* and *b1* show but can also be quite significant for the reasons explained above. Of particular interest is measurement *a3* with an overall DHCP

[17] Because we were testing with a set of different WLAN NICs we have selected WEP-based encryption as the greatest common denominator.

[18] we have also observed similar variability for DHCP in [15].

[19] We have also correlated vehicle speed as determined by GPS to our measurements and found the results to be in line with our above considerations.

[20] While our PCMP implementation is capable of autonomously dealing with changing interface parameters, observing such changes is only possible if PCMP is running on the device hosting the WLAN NIC.

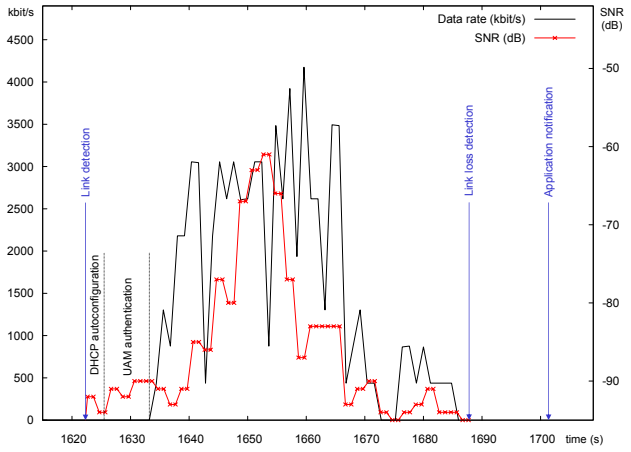

Figure 7: *Autobahn measurement with laptop setup at 120 km/h*

delay of 51 s, which was caused by driving behind a moving tram that was also driving eastwards and was additionally blocking the line of sight to the access point. For *b2* and *b4*, the access point becomes visible well east of the (red) traffic light TL5 but the association only completes between TL5 and TL4 when driving westward.

Against the end of the usage period, the data throughput goes down and reaches zero, usually *before* loss of the radio signal is noticed, thus mirroring the effect when entering the WLAN (so that simple link layer triggers on radio signal would be of little use for PCMP). For the urban notebook-only scenario in figure 8a, we further observe that link layer signal reporting continues for many seconds beyond the last transmitted data packet while the Linksys-based radio signal loss detection is much closer to the end of data transmission as can be seen from figure 8b. This may indicate that the Linksys hardware has better radio capabilities (more symmetric to the AP) so that beacon reception and frame transmission cease roughly at the same distance to the AP. This is subject to further study.

When we compare the performance of the laptop-only configuration to the embedded DT-MAG configuration based on the Linksys access point (based on figure 8), we can note the following: the maximum TCP throughput of the embedded DT-MAG configuration is approximately 2 to 3 times lower compared to the laptop-only configuration. We have been able to reproduce this performance difference (actually a factor of 3 to 4) in laboratory measurements where we have analysed the behaviour without the random effects caused by mobility on the road. Based on measurements from different test runs with different embedded DT-MAG configurations we have made the observation that the automatic WLAN selection and the WLAN scanning implies a significant performance penalty on the Linksys access point. Further analysis has shown that the Linksys access point's WLAN NIC exhibits a different a timing behaviour with respect to WLAN scanning, which has resulted in a higher scanning rate compared to the laptop-only configuration (one second interval compared to four second interval). We are currently revising the WLAN scanning implementation in order to accommodate different WLAN NIC implementations to improve the overall performance.

Our measurements with the embedded DT-MAG and the laptop-only configuration have also revealed that the Linksys access point-based embedded variant provides better WLAN radio communication performance: In most measurements, the coverage zone was significantly larger (obtained from corelating SNR readings and GPS coordinates) and generally the signal-to-noise ratio was also larger (also reflected by figure 8 although both configuration used the same external antenna.

Overall, our tests have shown that the stand-alone implementation of a Drive-thru Mobile Access Gateway is feasible and operates well in very different environments including an environment packed with non-accessible WLANs. Partly forced by the present constraints of the embedded access router platform, we have shown that the modular decomposition approach with message passing between local components is workable and has no noticeable performance impact as the individual components operate largely autonomously and the number of message exchanges is negligible. Offloading complex functionality from the embedded system (to the user device) even allows, e.g., potentially specialised access control functionality to be implemented where the knowledge about such WLANs resides and helps to avoid replicating these functions or even the associated credentials to a (shared) device that is not trusted from a user perspective.

With this, the DT-MAG system architecture exhibits the desired flexibility and allows functionality to migrate smoothly between user devices and the gateway.

Assuming a gateway with a WLAN interface implementation that is able to appear as multiple different WLAN cards simultaneously (or a hotspot that bases access control solely on IP addresses), this approach would allow even different users in the same vehicle to use different wireless service providers per hotspot at the same time. For example, current advanced WLAN access points provide the possibility to advertise multiple ESSIDs and to link these different network names to different RADIUS domains.[21] A similar mechanism could be used in a mobile gateway implementation where different ESSIDs represent differently configured DT-MAG services.

To enable such kind of operation, the the user must be able to dynamically locate an available DT-MAG, determine which functional range it offers (just WLAN bridging, supportive link status notification, full authentication, PCMP, application support). The user device must also obtain the Mbus configuration parameters for communication with the DT-MAG without manual interaction. A possible approach to this would be to employ service location and

[21] E.g., the Cisco Aironet 1100 series access points.

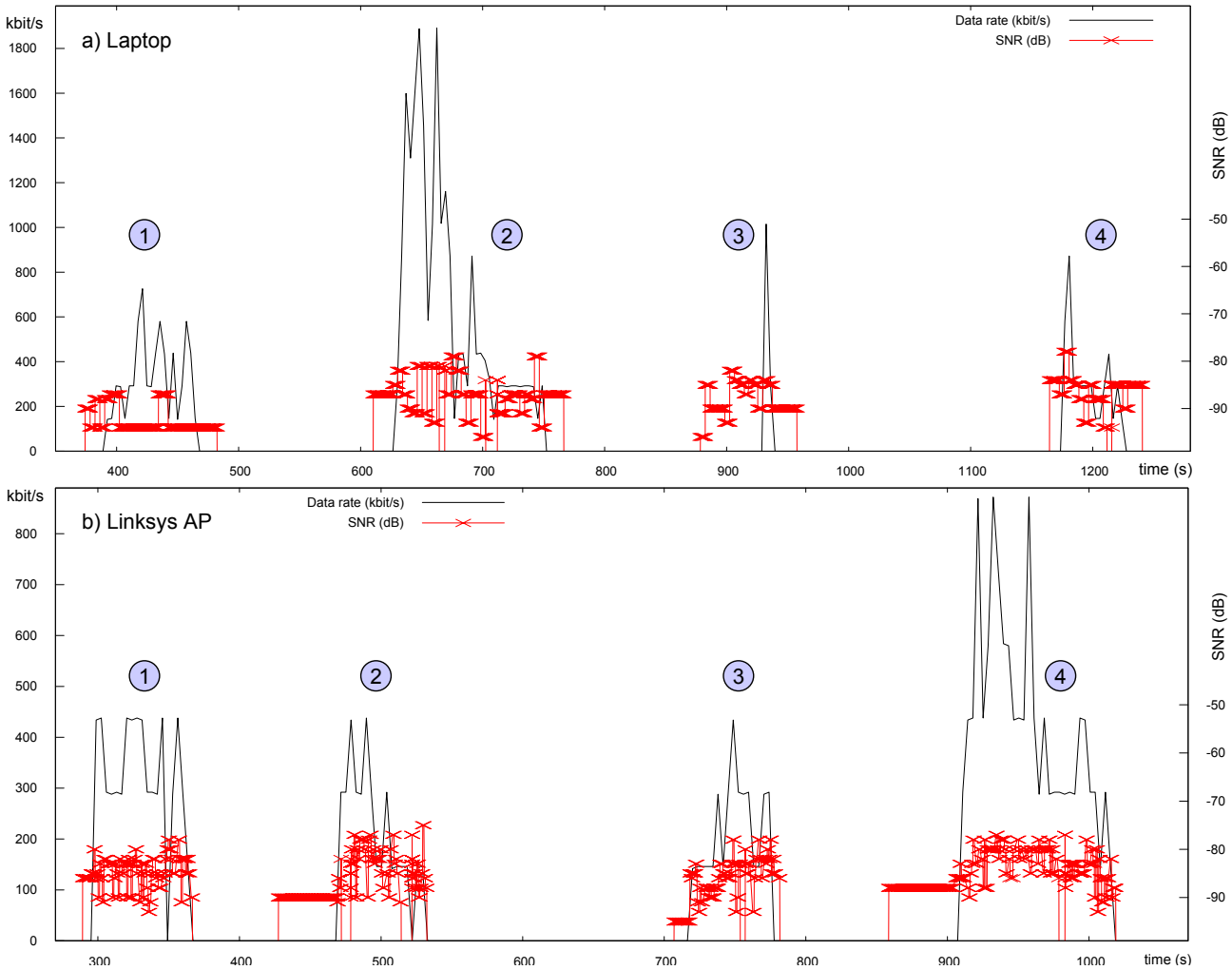

Figure 8: *Selected urban measurements with (a) laptop and (b) Linksys AP router setup at up to some 50 km/h (four passes)*

service association concepts such as the Dynamic Device Association (DDA) protocol [25]. Based upon MT-MAG capability information obtained in such a way, the mobile user may dynamically disable similar local services already provided by the DT-MAG and hence not needed on her device (which could also happen automatically).

8 Discussion: Gateway Location and Cross Layer Optimisations

The motivation for developing the DT-MAG system was to enhance the support for managing intermittent connectivity in scenarios, where the user device cannot be directly attached to the access link. This applies to some vehicular communication scenarios, where dedicated hardware is needed to establish connectivity but also to multi-user scenarios, where resource-sharing may be the primary motivation.

One interesting consequence of introducing such a gateway in intermittent connectivity scenarios is that the user device is no longer directly connected to the problematic link. Instead, all aspects of network access are taken care of by the gateway. This means that disconnections and varying link characteristics are not directly visible to higher layer protocols and applications. On the user device, the network link is no longer a volatile resource and the interface configuration remains constant.

When doing our first tests with the decomposed DT-MAG running the *ConnectivityDetector* and the *AutoAuthenticator* on the gateway and PCMP on the user device, we have immediately observed the different behaviour of PCMP. When directly connected to the access link, the PCMP module would notice the loss of connectivity quite quickly as the operating system's API would provide the corresponding trigger, e.g., by flagging an error condition for the socket. The PCMP function would then close the TCP connection and try to reopen it once the WLAN inter-

face becomes usable again.

Apparently, a different approach is required, when the DT-MAG is managing the access link—which is the motivation for introducing explicit indications about the link state to interested entities using the DT-MAG. This is also a structured and more robust way for higher layers and applications to learn about the link state. Instead of having to react to failures, e.g., a failed write operation on a UNIX socket descriptor, interested applications can be made aware of link characteristics. This allows a much better handling of intermittent connectivity and can also be helpful for simply adapting the application behaviour to changed network characteristics or for informing the user in a sensible way about the present connectivity status, e.g., by providing a hint in the user interface.

Of course, giving higher layers access to detailed link information can also be viewed as severe layer violation. In traditional Internet environments, it is desirable for application developers not having to deal with detailed information from lower layers but rather using a simple and abstract interface. However, our DT-MAG design offers the additional link state information as an optional service and thus allows applications that understand this information to operate much more efficiently. To date, this mainly affects our PCMP-based client-proxy and *new*, disconnection-tolerant applications. But even existing applications can benefit as they may use the improved PCMP-proxy that provides the intermittent connectivity management service.

Finally, the concept of using "lower layer" hints only for optimisations (rather than as a feature to rely upon) is indeed highly important when looking at a broader scope: While we are presently assuming a simple network topology within a vehicle that consists of a single link and is relatively reliable with a well-identified access link, the link experiencing intermittent connectivity may well be farther away or there may be several affected or unstable links. And, of course, the application should not need to know whether or not it is running on a mobile or a fixed node. Such assumptions may often become invalid in the future and, consequently, while optimisations may rely on them, the basic system operation should not.

9 Conclusion

This paper has presented the motivation for a DT-MAG and its modular design following the requirements and the grouping of functions we have derived from various operational scenarios. We have implemented two prototypes of the DT-MAG and have started testing them in real-world scenarios. The DT-MAG enables several users to efficiently share a common network access link and Drive-thru communication context. Its modular design and the link-local communication bus allow flexibly moving functionality between the DT-MAG and user devices. The design particularly supports users with devices capable of stand-alone Drive-thru operation to dynamically locate and take advantage of a DT-MAG.

While the motivation is to push as much functionality as possible to the DT-MAG to maximise its effectiveness, this requires common accounting and, to some degree, trust in the DT-MAG. Independent of these, however, Drive-thru functions running on a car router also imply that the persistent connection state resides on this router, thereby disallowing a user to resume communications initiated before entering and to continue after leaving the vehicle. While this is not an issue for devices built into, e.g., a bus, individual communication is clearly restricted. We are currently investigating whether simple PCMP state transfer mechanisms can be employed to mitigate this shortcoming. We are also looking into integrating application-specific functions with the car router in a way that works with PCMP in the router as well as on the clients. Finally, business aspects and their technical implications to make hotspots broadly available to Drive-thru users deserve further consideration so that Drive-thru Internet access becomes equally attractive for a single user in a car and anonymous individuals sharing resources on a bus or train.

Separating the user device from the access link in scenarios with intermittent connectivity by installing a gateway has interesting consequences as we have discussed in section 8. Our main conclusion from this discussion is that networking in challenging environments such as the Drive-thru-Internet environment mandates a different perspective on some acknowledged principles such as strict layering. Instead of shielding higher layers and applications from specific network characteristics, it can be useful, if not required, to distribute information about the path characteristics to endpoints. While in the DT-MAG case, the challenged access link is of main interest, we also observe that a system design should not make too restrictive assumptions about its operational environment, in particular the number and location of the critical link(s).

For future work, we are considering better support for the DT-MAG in making decisions about network selection and connectivity management, e.g., by integrating a *network information service* that allows the gateway to obtain detailed information about available networks in its environment. For mobile WLAN access, the network association could be supported by explicitly distributing hotspot properties such as ESSID, channel configuration, security mechanisms, but also hotspot operator(s), tariff schemes, and roaming options. For the Drive-thru-Internet environment, where the path of the vehicle can often be predicted (in cars, but especially in trains) this could significantly improve the DT-MAG performance.

Furthermore, we are investigating additional hardware platforms for the embedded DT-MAG system, as the current platform is not able to host the complete functionality for managing intermittent connectivity (we are also looking into optimising our prototype implementation). In that direction, we will also experiment with different local net-

work interfaces, e.g., offering local WLAN access to the DT-MAG, and we would like to enhance the DT-MAG behaviour by employing multiple external WLAN interfaces, e.g., in order to concentrate the scanning function on one interface but also to enhance the performance by means of data striping. Automatically coordinating between several DT-MAGs mounted alongside a vehicle (e.g., at both ends of a train) while maintaining the modularity concept is another interesting research aspect.

For the long-term, this could be further generalised to heterogeneous networks, e.g., employing 3G in addition to WLAN(s), so that the concepts of *always best connected* and handling intermittent connectivity are combined. Allowing to relax the connectivity requirements will add flexibility and thus add choice for the user: choice when to use which networks and even allow for service interruptions so that, besides availability, cost and efficiency of communication networks gain importance in mobile communications. With enhanced user policies, a user will be able to better specify her preferences and thus emphasise the *best* in "always best connected"—in the sense of *most appropriate* from her perspective—as opposed to the *always*.

10 Acknowledgements

The authors would like to thank Mark Koch for the DT-MAG implementation, Dirk Meyer for the PCMP implementation, and both of them for the measurements.

References

[1] E. Gustafsson and A. Jonsson. Always Best Connected. *IEEE Wireless Communications*, 10(1):49–55, February 2003.

[2] G. Leijonhufvud. Multi access networks and Always Best Connected, ABC. MMC Workshop, November 2001.

[3] P. Rodriguez, R. Chakravorty, J. Chesterfield, I. Pratty, and S. Banerjee. MAR: A Commuter Router Infrastructure for the Mobile Internet. In *Proc. of ACM Mobisys*, June 2004.

[4] M. Zitterbart et al. IPonAir – Drahtloses Internet des nächsten Generation. PIK, Vol 26, No 4, October 2003.

[5] B. Anton, B. Bullock, and J. Short. Best Current Practices for Wireless Internet Service Provider (WISP) Roaming, Version 1.0. Wi-Fi Alliance, February 2003.

[6] A. Balachandran, G. M. Voelker, and P. Bahl. Wireless Hotspots: Current Challenges and Future Directions. In *Proceeding of WMASH 2003*, September 2003.

[7] J. Ott and D. Kutscher. Exploiting Regular Hot-Spots for Drive-thru Internet. In *Proceedings of KiVS 2005, Kaiserslautern*, March 2005.

[8] J. Ott and D. Kutscher. Why Seamless? Towards Exploiting WLAN-based Intermittent Connectivity on the Road. In *Proceedings of the TERENA Networking Conference, TNC 2004, Rhodes*, June 2004.

[9] J. Ott and D. Kutscher. Drive-thru Internet: IEEE 802.11b for „Automobile" Users. In *Proc. of IEEE Infocom, Hong Kong*, 2004.

[10] J. Ott and D. Kutscher. The "Drive-thru" Architecture: WLAN-based Internet Access on the Road. In *Proc. of VTC Spring 2004*, May 2004.

[11] A. Baig, M. Hassan, and L. Libman. Prediction-based Recovery from Link Outages in On-Board Mobile Communication Networks. In *Proceeding of IEEE Globecom 2004*, December 2004.

[12] Website of FleetNet. http://www.fleetnet.de/, 2003.

[13] L. Bononi, M. Conti, and E. Gregori. Design and Performance Evaluation of an Asymptotically Optimal Backoff Algorithm for IEEE 802.11 Wireless LANs. In *33rd Hawaii International Conference on System Sciences*, January 2000.

[14] J. Ott and D. Kutscher. A Disconnection-Tolerant Transport for Drive-thru Internet Environments. In *Proc. of IEEE Infocom, Miami*, 2005.

[15] J. Ott, D. Kutscher, and M. Koch. Towards Automated Authentication for Mobile Users in WLAN Hot-Spots. In *Proceedings of VTC Fall 2005*, September 2005.

[16] M. Esbjörnsson, O. Juhlin, and M. Östergren. The Hocman Prototype - Fast Motor Bikers and Ad-hoc Networking. Proc. of MUM, 2002.

[17] N. Fuke, H. Izumikawa, K. Sugiyama, and M. Nohara. Development of CDMA2000 1xEV-DO/Wireless LAN Gateway and Performance Evaluation in Static/Mobile Environment. Technical Report 553, IEICE, January 2005.

[18] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert. Network Mobility (NEMO) Basic Support Protocol. RFC 3963, June 2005.

[19] C.-W. Ng, E. K. Paik, T. Ernst, and M. Bagnulo. Analysis of Multihoming in Network Mobility Support. Internet Draft draft-ietf-nemo-multihoming-issues-03.txt, Work in Progress, July 2005.

[20] T. Ernst, K. Uehara, and K. Mitsuya. Network mobility from the internetcar perspective. In *Proceedings of the International Conference on Advanced Networking and Applications*, March 2003.

[21] K. Mitsuya, K. Uehara, and J. Murai. The in-vehicle router system to support network mobility. In *ICOIN2003 Proceedings*, February 2003.

[22] K. Fall. A Delay-Tolerant Network Architecture for Challenged Internets. Proceedings of ACM SIGCOMM 2003, Computer Communications Review, Vol 33, No 4, August 2003.

[23] M. Bechler, L. Wolf, O. Storz, and W. Franz. Efficient Discovery of Internet Gateways in Future Vehicular Communication Systems. In *Proc. of VTC Spring 2003, Jeju, Korea*, April 2003.

[24] J. Ott, C. Perkins, and D. Kutscher. A Message Bus for Local Coordination. RFC 3259, April 2002.

[25] D. Kutscher and J. Ott. Dynamic Device Access for Mobile Users. In *Proceedings of the 8th Conference on Personal Wireless Communications*, 2003.