# Mobile Loading of the GeldKarte: Performance Optimization within Mobile Communication Networks

Guido Zavagli[1], Ian Herwono[2], Ralf Keller[1]

[1]Ericsson Eurolab Deutschland GmbH
Ericsson Allee 1, 52134 Herzogenrath, Germany
{guido.zavagli, ralf.keller}@eed.ericsson.se

[2]Communication Networks, Aachen University of Technology
Kopernikusstraße 16, 52074 Aachen, Germany
ian@comnets.rwth-aachen.de

*Abstract*

Mobile users will increasingly demand access to Electronic Commerce services. Within this paper, we investigate the adaptation of state-of-the-art Electronic Commerce technologies to mobile networks. Our focus is put on the integration of smart card based Electronic commerce technologies, namely on the German GeldKarte system, into the GSM system. An overview on both the GeldKarte system and on GSM messaging services is presented, followed by a discussion of the integration of the GeldKarte loading functionality into the mobile system. A detailed description of the mapping of GeldKarte transactions to specific GSM messaging services is provided, combined with a performance evaluation of this integration. Also an outline on the relation of our proposal to the upcoming WAP standard is provided.

## 1. Introduction

Mobile users will increasingly demand access to Electronic Commerce services. Therefore the possible adaptation of existing electronic commerce technologies to mobile networks has to be investigated. Mobile Electronic Commerce (MEC) is for us a general concept covering any business transaction executed electronically between at least two parties. At least one of these parties is mobile and uses a wireless transmission medium at least on the first link for the communication with the other parties [6].

Smart Cards are used today in many Electronic Commerce applications and are small enough to be carried around while being mobile [8]. The EC-GeldKarte system is also based on the Smart Card technology. It was developed with stationary devices in mind. The EC-GeldKarte has two main functions: payment and loading. Both payments and load transactions are nowadays performed using fixed (stationary) devices; in the case of loading this device is called a load terminal. Load terminals are usually placed within banks or are co-located with automatic teller machines (ATM). Both solutions imply that users running out of money on their EC-GeldKarte have to find such a load terminal. To improve user satisfaction with the EC-GeldKarte system, we therefore investigate the integration of the load functionality of the EC-GeldKarte system into mobile devices, especially for the GSM system.

This paper presents in section 2 an overview on the German EC-GeldKarte system. In section 3 we describe in detail how loading of the GeldKarte can be integrated into the GSM system. Special emphasis is put on the mapping of GeldKarte transactions to specific GSM data services. The performance evaluation of the proposed integration into the GSM system is described in section 4. In section 5 we outline how the GeldKarte system can be integrated within the upcoming WAP architecture. This paper concludes with an outlook.

## 2. German EC-GeldKarte

The EC-GeldKarte system has been introduced in 1997 as the first electronic purse system in Germany. The electronic purse is represented by a chip-card, which consists of a complete highly integrated micro-controller, in contrary to ordinary health cards or phone cards. The chip integrates a processor, memory and a card specific operating system.

Two types of GeldKarte are defined, **the bank account linked GeldKarte** with a PIN, and the **"white" GeldKarte** without linked user bank account and without PIN. The PIN is used for user authentication.

Several transactions are defined within the EC-GeldKarte system for loading the GeldKarte and for performing payments. These are described in more detail in the following section.

## 2.1. Transaction Overview

As illustrated in Figure 2.1, the transactions available when using the GeldKarte with linked user bank account, including the withdrawal and payment transactions, are briefly described:

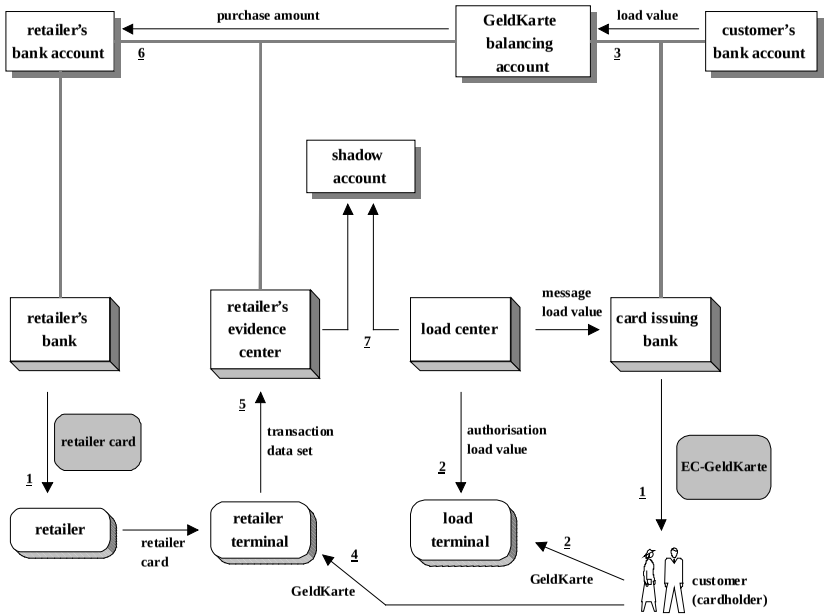



**Fig. 2.1**: Architecture of the German GeldKarte System

1. The customer acquires a GeldKarte from the bank, with which he has the account. The retailer acquires also a chip-card (retailer card) from his bank. The retailer card is installed on the retailer terminal.
2. The customer can load an amount of money (up to 400 DEM) to the GeldKarte at an appropriate load terminal.
3. The loaded value plus fee is charged from the customer's bank account by the card issuing bank, and credited to the **special GeldKarte balancing account** (*germ. Börsenverrechnungskonto*).
4. During a payment transaction the purchase amount is charged from the customer's GeldKarte, i.e. the stored value is decreased. The retailer card generates a certified transaction data set consisting of the purchase amount, the customer's card identity and the date and time of the transaction. This transaction data is temporarily stored on the retailer terminal system.
5. The transaction data set is transferred via an appropriate network (e.g. phone network) to the evidence center chosen by the retailer on a periodical basis (e.g. daily basis).
6. The evidence center credits the purchase amount onto the retailer's bank account. The purchase amount is then charged from the customer's GeldKarte balancing account.
7. Additionally, a shadow account is managed and actualized for each transaction performed by the GeldKarte.

## 2.2. The Load Transaction

The GeldKarte can only be loaded after an on-line authorization, which checks that the value to be loaded can be credited to the GeldKarte balancing account of the cardholder. The authorization is done by the load center, which can be identified using the card's serial number. Two types of load transaction can be distinguished:

- loading the value from the cardholder's bank account and
- loading the value against other payment means (e.g. cash).

During a load transaction, the load terminal acts as an intermediate device between the card and the load center, and provides the user interface to the cardholder.

Load transactions from the cardholder's bank account are only possible for account linked GeldKarte after entering the card specific PIN. During the on-line authorization the load center proves the availability of the value to be loaded from the cardholder's bank account directly, and performs the value transfer from the bank account to the card specific GeldKarte balancing account. The information about the bank identifier (BLZ) and the account number, which are stored in an encrypted form on the card, will be transferred to the load center during each on-line load transaction. An unauthorized change of this information during the load transaction leads to a

rejection of the load request. This ensures that always the same bank account will be charged according to the card.

In order to secure the point-to-point communication between the card and the load center, the messages to be exchanged are attached with a **MAC** (Message Authentication Code), which is previously generated using a shared dynamic secret key. A sequence number shared by both parties and a repeat counter are used to generate the MAC. This way both parties can perform a sequence control of the exchanged data.

Loading the card value against other payment means is possible for bank account linked cards as well as for cards without bank account and PIN. During this load transaction the communication will be also secured with MAC using the sequence number and the repeat counter.

In this load transaction type, first the load terminal provider must be convinced that the load value is already paid to him, e.g. through cash payment. The load value will be always charged from the provider's bank account and will be credited to the GeldKarte balancing account, according to the card serial number. If a GSM network operator can act as the load terminal provider, the load value plus fee can be charged from the customer's (subscriber's) bank account.

When the load terminal receives the confirmation message to load the value onto the card, it initiates an on-line authorization request using the bank identifier (BLZ) and the provider's account number. The load center charges the loaded value from the given provider's account and credits it to the card's balancing account.

Account linked GeldKarte can also be unloaded. The load center adjusts the value stored on the card to zero and transfers the previously stored value from the card's balancing account back to the cardholder's bank account.

## 3. GeldKarte Load Transaction Service in GSM System

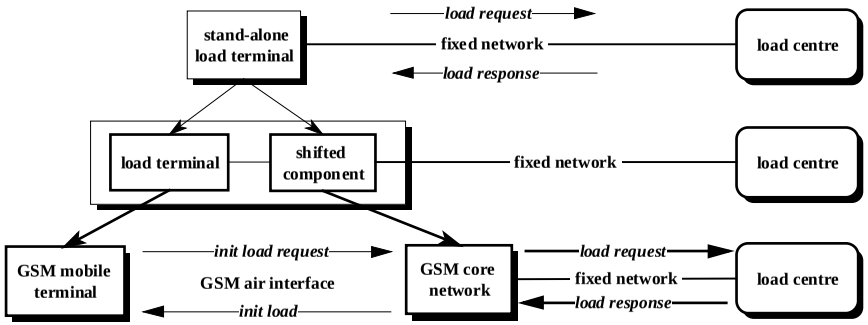
One objective of the integration of the GeldKarte load transaction scheme into the GSM system is to minimize the size of the transaction data to be transferred over the air interface without decreasing the transaction security level.

### 3.1. System Scenario

A cardholder can load his electronic purse by using specific load terminals commonly provided by banks. A load terminal is connected with a load center doing the corresponding online-authorization of the load transactions.

Within a load transaction, two messages are exchanged between the load terminal and the load center: *load request* and *load response*. The load request message is sent by a load terminal to the load center. The load center replies with a load response message – this message is used to load the electronic purse with the authorized amount of money.

According to GeldKarte technical specification [1] both messages must conform to the ISO 8583 message format. However the GeldKarte specification allows the use of an internal format for both messages when the (stand-alone) load terminal is split into two main components: the load terminal and its shifted component. As depicted in figure 3.1, the load terminal may be implemented in a GSM mobile station (*mobile load terminal*) and the shifted components in one or several nodes within the core network, e.g. in the HLR.




: Splitting of the GeldKarte load terminal

### 3.2. Format of the Internal Messages

During a load transaction, two transaction messages using the internal format must be exchanged via the radio interface between the mobile load terminal and the core network. These messages are called *init load request* and *init load*. The format and the length of each data element included in the messages are determined by the GeldKarte specification [1] and cannot be modified arbitrarily. In case of the request message the core network is responsible for

adding the proper data elements necessary for generating an ISO 8583 compliant load request for the load center. Consequently the core network has to convert the ISO 8583 formatted load response message originating from the load center into an internal message to be transferred to and processed by the mobile station in order to load the electronic purse.

The *init load request* message, which is 118 bytes of length, contains the following data elements:

- *type* (1 byte): the type of the message, i.e. request,
- *transaction amount* (3 bytes): the load amount to be charged from card's bank account,
- *trace number* (3 bytes): identification of a load transaction being performed,
- *transaction time* (2 bytes): the time at which the request message is generated,
- *transaction date* (3 bytes): the date on which the request message is generated,
- *condition code* (1 byte): type indication of the load terminal generating the request message,
- *terminal ID* (8 bytes): a unique terminal identification,
- *encryption parameter* (9 bytes): parameter used to generate cryptographic session key,
- *data field EF_ID* (22 bytes): encrypted identification data of the electronic purse secured with a message authentication code (MAC),
- *answer data to initiate loading* (58 bytes): the card response data secured with a MAC to initiate load transaction (generated during transaction initialization),
- *MAC* (8 bytes): a cryptographic checksum generated over the preceding 110 bytes of the request message.

The core network converts the *init load request* message to an ISO 8583 formatted *load request* message (214 bytes) and forwards it to the load center.

The ISO 8583-formatted *load response* message of the load center, which is up to 169 bytes in length, will be converted by the core network into an *init load* message. The converted response message is 60 bytes in length and contains the following data elements:

- *type* (1 byte): the type of the message, i.e. response,
- *answer code* (1 byte): indication of the transaction status, i.e. succeeded or failed,
- *condition code* (1 byte): indication of a required change of the amount limits determined for the electronic purse,
- *load command message* (43 bytes): a part of the command message of the load center to be used for loading the electronic purse (secured with MAC),
- *rest amount* (6 bytes): the rest limit of the card's load amount,
- *MAC* (8 bytes): a cryptographic checksum generated over the preceding 52 bytes data elements.

## 3.3. Transaction Security

The transaction security is ensured since specific information from the electronic purse such as the bank account number is included in the data field EF_ID, which is transferred over the radio interface in an encrypted format. Furthermore the integrity of several data elements and the whole internal messages is ensured by message authentication codes (MACs). The data encryption and the MAC generation are performed using the symmetric DES (Data Encryption Standard) or Triple-DES encryption algorithm.

## 3.4. Service Implementation

SMS (*Short Message Service*) and USSD (*Unstructured Supplementary Service Data*) are considered to be used as the data bearer services for the provision of the German GeldKarte load transaction service in GSM networks. Below both system approaches are presented with regard to the format and the logical flow of the messages to be exchanged between the involved network nodes.

### 3.4.1 SMS Approach

Point-to-point SMS, which operates on GSM signaling channels, enables the exchange of messages with up to 140 bytes between SMEs (Short Message Entities) or between an SME and the SMSC (Short Message Service Center). The service center is responsible for receiving and forwarding short messages in a store-and-forward mode to the addressed recipients independent of their current locations. The receipt of each short message must be confirmed by the mobile station or the service center by means of a so-called delivery report.

In order to provide the GeldKarte load transaction service using SMS the mobile station as a mobile load terminal will act as an SME and the functionality of the load terminal's shifted component is implemented in another SME as an external service node (ESN) with direct connection to the SMSC. Since both internal transaction messages (*init load request* and *init load*) are less then 140 bytes in length, each of them can be exchanged as a single short message between the mobile station and the ESN through several GSM network nodes (BSC, MSC, HLR, etc.). The request is transferred as a mobile originated short message (MO-SM) and the response as a mobile terminated short message (MT-SM). Figure 3.2 depicts the logical message flow within a successful load transaction whereby the electronic purse (i.e. the GeldKarte) is inserted in an additional chip-card reader of the mobile station.

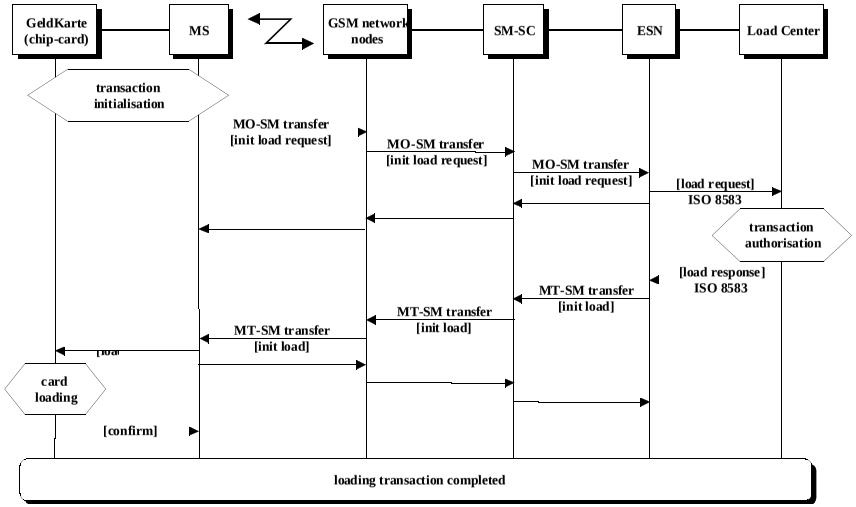



**Fig. 3.2**: SMS message flow of a successful load transaction

### 3.4.2 USSD Approach

USSD provides a mechanism to exchange unstructured messages as USSD strings between a mobile station and a specific USSD application implemented in a network node. In contrast to SMS it is possible to transfer several messages within an established USSD dialogue. The main objective of USSD is to facilitate the creation of operator-specific services.

Within a GSM network supporting USSD, there exist several USSD handlers and one or more USSD applications, as depicted in figure 3.3. USSD handlers are responsible for forwarding USSD messages to the addressed USSD application. USSD messages may be initiated by mobile users (mobile initiated USSD) or by the core network (network initiated USSD).

A USSD message consists of a USSD string and a DCS (Data Coding Scheme), which contains an alphabet and a language indicator. The alphabet indicator indicates how the USSD string should be interpreted, and the language indicator indicates the language used. The USSD-DCS is formatted according to the *Cell Broadcast DCS* [3][5]. The maximal length of the USSD string is not clearly defined in GSM specification. According to GSM 03.38 [3] the USSD string is limited to 82 octets.

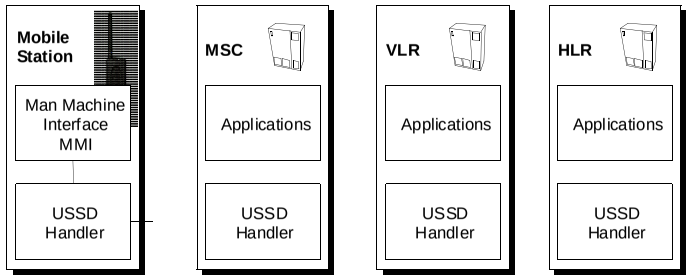



**Fig. 3.3**: USSD handling in GSM networks

Furthermore a USSD message contains a service code (SC) through which the addressed USSD application as well as the network node in which the corresponding USSD application is implemented can be identified.

Assuming that a USSD message can contain up to 82 octets of USSD string, the internal load transaction message *init load request* with 118 bytes of length can not be transferred within one USSD message. Hence an additional transport protocol for USSD is necessary: USSD Message Transfer Protocol (UMTP). The transport protocol should operate at the application layer level - in addition to the GSM data link layer protocol ($LAPD_m$) - and is implemented both on mobile stations and the core network as a part of the corresponding USSD application. Its main function is to initiate and to ensure the

segmentation of internal transaction messages into several appropriate USSD messages at the mobile side and their reassembly at the network side, as shown in figure 3.4.

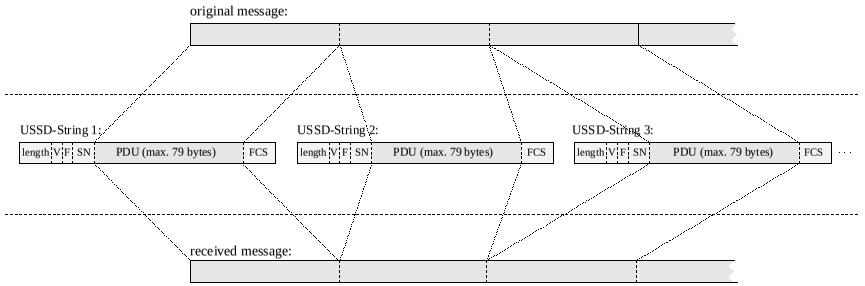



**Fig. 3.4**: Message segmentation by the additional transport protocol

The additional transport protocol splits - if necessary - the internal messages into message segments (*PDUs - Protocol Data Units*) each with a maximal length of 79 bytes. In order to identify the segments the following information is assigned to each PDU:

- a 5-bit sequence number (SN), hence a message can be split into 32 USSD messages maximally,
- a 2-bit number to represent the current protocol version (V),
- one flag-bit (F) to indicate the last message segment,
- an 8-bit number to represent the total length of the internal message (user data),
- an 8-bit checksum FCS (Frame Check Sequence) to detect transfer errors.

When the mobile station intends to transfer an internal message in form of a *USSD request* to the core network, it forwards the message to the mobile-side (additional) transport protocol. The protocol is then responsible to split the message - if necessary - and to send the appropriate USSD messages, as well as to receive the corresponding network response(s). The first message segment will be transferred as a *USSD request* to the network-side transport protocol, which in turn checks the message and responds with another *USSD request* to request the next message segment. All the following message segments will be sent by the mobile station as *USSD responses*, because the mobile station can only send one *USSD request* within one USSD dialogue [8]. After the last message segment is detected the original message will be reassembled and forwarded to the network-side USSD application. Note that due to specific characteristics of the USSD mechanism, it is not possible for the mobile device to send a USSD message before the acknowledgement for the previous USSD message has been received successfully.

Figure 3.5 presents the logical message flow of a successful GeldKarte load transaction within one USSD dialogue. The network-side USSD application which implements the functionality of the load terminal's shifted component is allocated in an external service node (ESN) connected to the HLR.

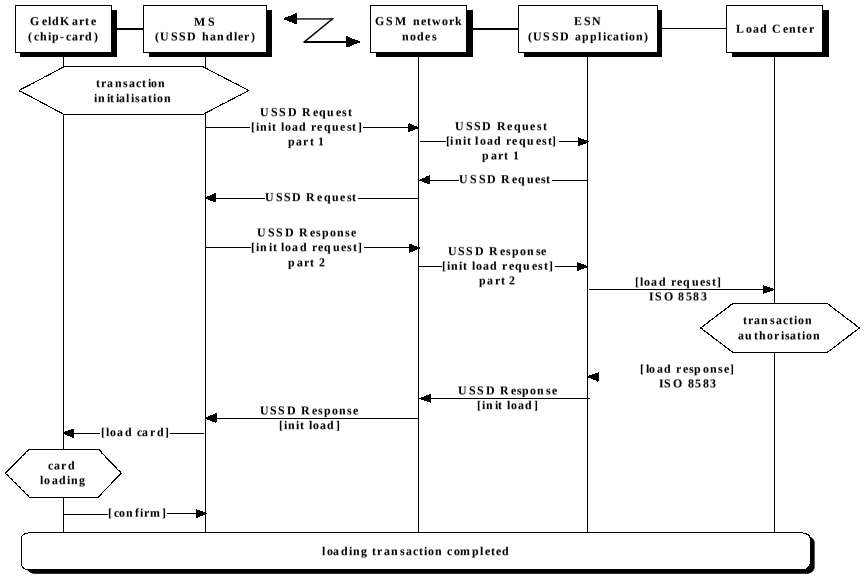



**Fig. 3.5**: USSD message flow for a successful load transaction

## 4. Performance Evaluation

The main objective of the performance evaluation described in this section is to

investigate the timing behavior of the GeldKarte load transaction. For this purpose, all network components and transport mechanisms involved in load transactions must be modeled.

## 4.1. Performance Evaluation Models

In general, the performance evaluation of telecommunication networks such as the GSM system can be performed with two different approaches, either by using an analytical model, or by performing simulations. For this study, the analytical approach was used. The corresponding models are described in this section (a detailed description can be found in [9]).

### 4.1.1 Basic User Model

The user model describes the behavior of the GSM subscribers. For each standard GSM service (call, hand-over, etc.), the traffic intensity caused by a subscriber with an active mobile state (BHCA, Busy Hour Call Attempts) is defined. These are used to define the background load of the GSM system. The traffic values chosen here correspond to the GSM PLMNs found in Germany mid 1997.

### 4.1.2 Network Structure Model

The network model models the nodes and signaling links of the GSM system as shown in figures 3.2 and 3.3. It is assumed that the HLRs and MSCs are co-located, i.e., implemented in the same network node, and that almost each MSC acts also as GMSC or IWMSC. The external service node (ESN) on which the USSD application can be implemented is connected with the GSM system over one or several signaling transfer points (STPs), which act as routers. The nodes and links are shown in figure 4.1.

## 4.2. Service Analysis

In order to analyze the timing behavior of the GeldKarte load transaction via GSM and its impacts on existing GSM networks, the complete message flow of this service including signaling messages has to be developed. Figure 4.2 shows the corresponding message flow for the USSD approach, whereby the network-side USSD application is implemented in an ESN.

The relevant delays and waiting/processing times within a USSD based load transaction are presented in figure 4.3. In case of an SMS channel allocation, initialization and release procedures must be performed twice within a load transaction.

For each of the operations shown in figure 4.3, the corresponding analytical model is described in [9].

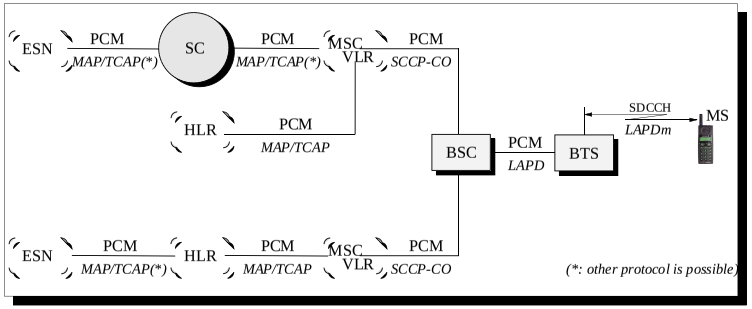



**Fig. 4.1**: Protocols and links used for USSD and SMS

### 4.2.1 Total Transaction Time

Taking all delays together, we are now able to calculate the total duration of a load transaction (not the dialogue length) applying the USSD and SMS mechanisms. The results are shown in figure 4.4 for a usage ranging from 0 to 0.2 BHCA (represents approximately one fourth of normal call setup operations). The USSD based load transaction with 27 seconds of transaction time is about one second faster than the SMS approach. Furthermore, the load transaction time increases only slightly in case of higher service traffic intensity.

### 4.2.2 Impact on the GSM system

The introduction of new services causes additional load in nodes and links of the GSM network and thus impacts other existing services. Figure 4.5 shows the additional load on network (signaling) nodes (BSC, MSC, HLR, and ESN) caused by the USSD GeldKarte application implemented in an external node. The major impact can be noticed on the external node since only one external node is used for the GeldKarte service. The impacts on the HLRs are also

noticeable. Usually load on HLRs is caused by such operations like requests on routing information. In case of the GeldKarte service all HLRs must process all USSD operations of the external node. The impacts on other signaling nodes are minor since the service traffic is distributed on several nodes.

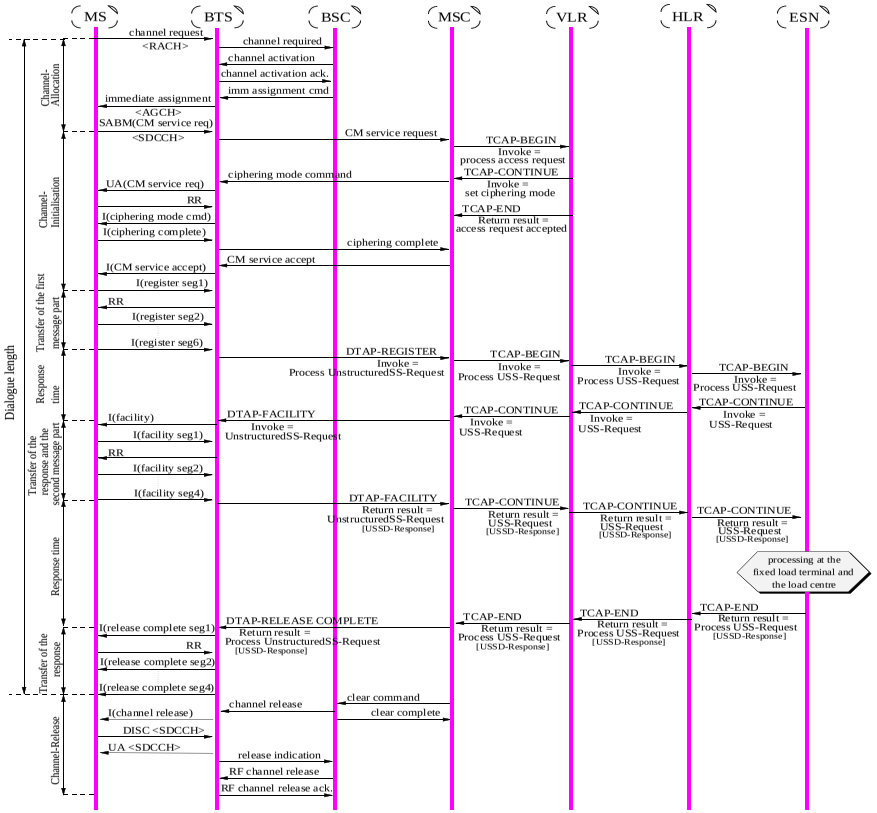
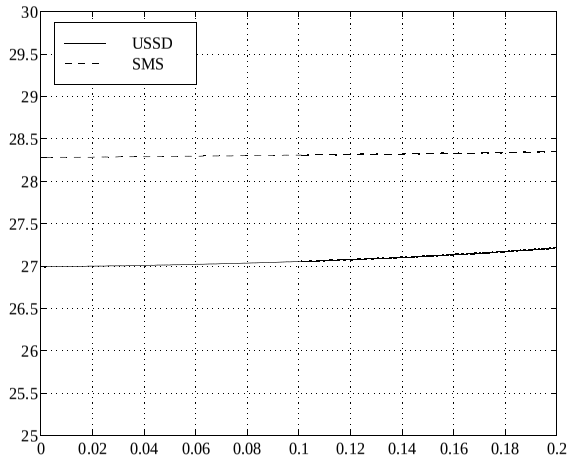



**Fig. 4.2**: Complete message flow of a USSD based load transaction



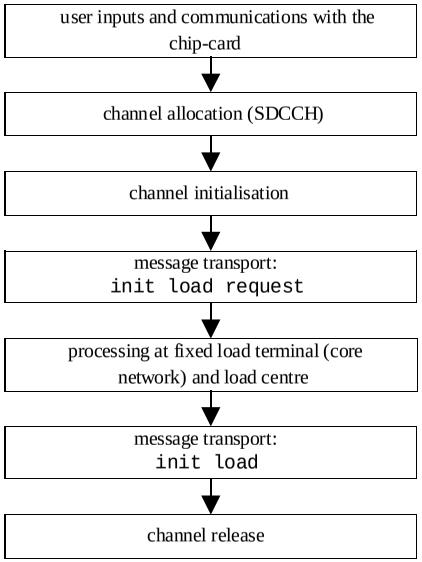
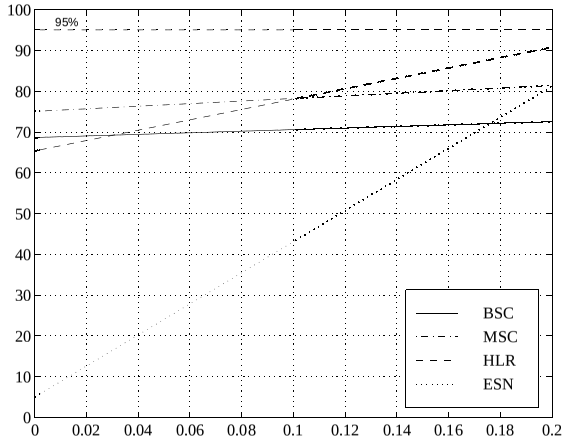
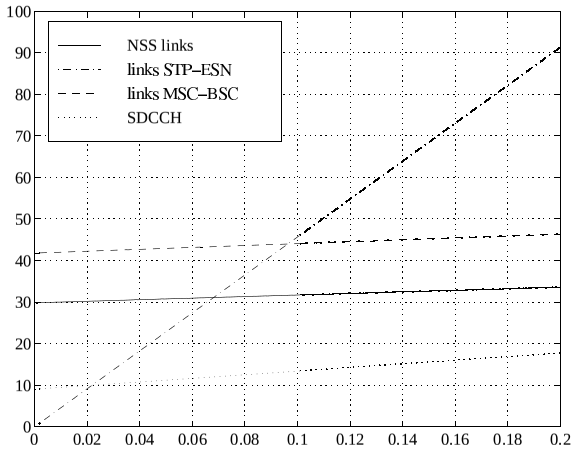




**Fig. 4.3**: Processing times and delays within a USSD based load transaction

Figure 4.6 shows the impact of the GeldKarte service on signaling links. Major impacts are experienced on the links between HLRs and the external node. Impacts on remaining links are minor.




Unwired Planet's Handheld Device Markup Language (HDML) and Handheld Device Transport Protocol (HDTP), Ericsson's Intelligent Terminal Transfer Protocol (ITTP), and Nokia's Smart Messaging and Tagged Text Markup Language (TTML). At present a number of WAP technical specifications [7] are available for public review.

### 5.1.1 Protocol Architecture

The protocols specified in WAP are suited to the characteristics of radio interface such as narrow bandwidth, unstable connection, small data packets or

high latency, and for efficient use of device resources such as low memory or CPU usage. According to its architecture, various wireless communication systems (GSM900, GSM1800, CDMA system), transport mechanisms (GPRS, SMS, USSD) and mobile terminal types are supported by WAP. The following mandatory architectural components are required in a WAP compliant system (Fig. 5.1):

1. Wireless Application Environment (WAE) incl. WML (Wireless Markup Language) browser and WMLScript interpreter
2. Wireless Session Layer (WSP)
3. Wireless Transport Layer Security (WTLS)
4. Wireless Transport Layer (WTP/D, WTP/T, WTP/C)
5. Bearers (SMS, USSD, GPRS, CDMA, etc.)

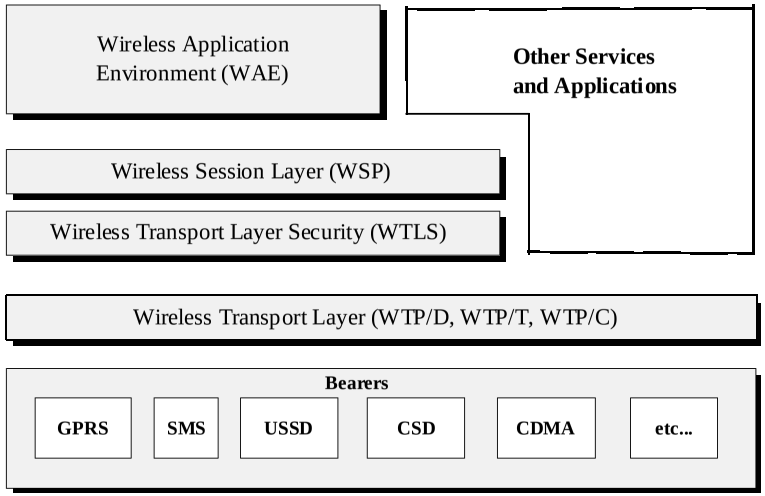



**Fig. 5.1:** WAP architecture

### 5.1.2 System Scenario

In general, the WAP architecture is based on the client/server programming model applied by Internet-WWW (World Wide Web). As shown in Figure 5.2, the basic concept of WAP is to filter or to convert common Internet pages written in HTML (HyperText Markup Language), into WML formatted pages, which can then be displayed by WAP compliant wireless terminals. Another possibility is to implement or to 'write' the services (TeleVAS) in WML or WMLScript.

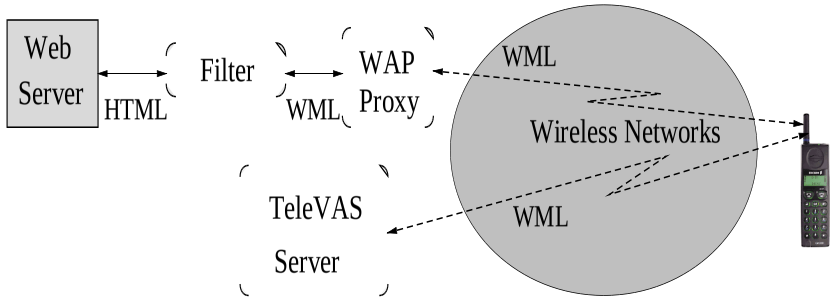



**Fig. 5.2:** The WAP system scenario

WML is a tag-based document language, optimized for specifying presentations and user interfaces on small screens of narrow-band (wireless) devices. WMLScript is a lightweight procedural scripting language based on JavaScript™ (ECMAScript) [2]. It can be used together with WML to provide intelligence to the clients such as checking the validity of user input or accessing the SIM (Subscriber Identity Module) card inserted in a GSM mobile station. WMLscript enables the generation of messages and dialogs locally thus reducing the need for time consuming round trips to the network server.

Appropriate WAP layers ensure the reliability and the security of the data communications between the involved entities (WAP clients and servers) over the radio interface. Furthermore, the end users should be presented with a consistent and vendor controlled application man machine interface.

## 5.2. WAP for the GeldKarte Load Transaction Service

Concerning its features and characteristics, WAP provides a good basis for electronic commerce services such as the GeldKarte load transaction service within GSM networks through the Internet. By using the WMLScript capability the same approach introduced in section 3.1 (splitting of the load terminal) can be implemented within WAP system environment, as depicted in figure 5.3. Because WAP is designed for general Internet based services, the WAP

approach of the GeldKarte transaction service can be expected to lead to longer transaction time than the proposed proprietary solution, e.g. the USSD approach, due to additional protocol overheads. The technical implementation of the WAP approach and the corresponding performance evaluation are subjects of further study.

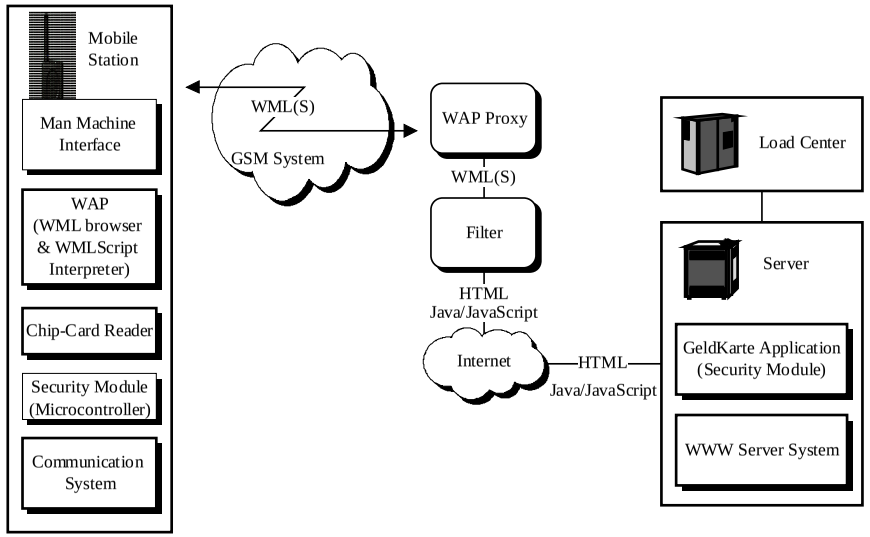



**Fig. 5.3**: Script-based WAP approach of the GeldKarte load transaction service

## 6. Conclusion

Chip-card based payment systems are already on the market. They are gaining more and more momentum, driven by both bank and merchants. Within this paper, we have described how loading of the German EC-GeldKarte system can be optimized within the GSM system. We have described in detail how the different transactions can be mapped on SMS and USSD messages. The performance evaluation of the proposed integration into the GSM system has shown how advantageous our approaches are. Nevertheless, the most optimal solution is not always the best solution to implement, therefore we have outlined how EC-GeldKarte loading can be implemented within the upcoming WAP architecture.

Payment with the EC-GeldKarte is only possible when money has been loaded to it. Mobile loading of the EC-GeldKarte is therefore one important step to increase user satisfaction and to support the promotion of the system into all-day practice. Important questions not tackled within this paper are how to communicate with chip-cards while being mobile. Chip-card readers are already integrated into mobile phones, but used for the SIM card within GSM phones. Second card readers are difficult to be integrated into small phones and increase production costs. We have to seek for more advanced solutions, easy to handle and cheap to sell.

## 7. References


[1] "Interface Specification of EC-chipcards, version 2.2 (germ.)", Bank-Verlag GmbH, Deutscher Genossenschaftsverlag, Deutscher Sparkassenverlag, VöB-Zahlungsverkehrsdienstleistungs-Gesselschaft (1997).

[2] ECMA, "Standard ECMA-262: „ECMAScript Language Specification„," (1997).

[3] ETSI, "GSM recommendations 03.38: Alphabets and language-specific information", (1996).

[4] ETSI, "GSM recommendations 03.90: Unstructured Supplementary Service Data (USSD) - Stage 2" (1996).

[5] ETSI, "GSM recommendations 09.02: Mobile application part (MAP) specification", (1996).

[6] Keller, R., G. Zavagli, J. Hartmann, and F. Williams, "Mobile Electronic Commerce: GeldKarte Loading Functionality in Wireless Wallets", in: Proceedings of IFIP Electronic Commerce 98, Hamburg (1998).

[7] WAP-Forum, Technical Documentation of WAP, `http://www.wapforum.org/ docs/technical.htm.`

[8] Wrona, K. R. Keller, I. Herwono, and F. Williams, "Electronic Commerce in Deutschland und weltweit: Neue Zahlungssysteme und ihre Anwendungsbereiche", *PIK*, 21(1):3—10 (1998).

[9] Zavagli, G., "Performance evaluation of operator specific service provisioning within the cellular GSM mobile radio network", Master thesis, RWTH Aachen, Communication Networks (1997).