

Zeitschrift: Bulletin Electrosuisse
Herausgeber: Electrosuisse, Verband für Elektro-, Energie- und Informationstechnik
Band: 97 (2006)
Heft: 7

Artikel: Plattformunabhängige mobile Software
Autor: Brandt, Peter K.
DOI: <https://doi.org/10.5169/seals-857659>

Nutzungsbedingungen

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. [Siehe Rechtliche Hinweise.](#)

Conditions d'utilisation

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. [Voir Informations légales.](#)

Terms of use

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. [See Legal notice.](#)

Download PDF: 14.05.2025

ETH-Bibliothek Zürich, E-Periodica, <https://www.e-periodica.ch>

Plattformunabhängige mobile Software

Mobiles Zugpersonalgerät der SBB

Die Anzahl mobiler Geräte ist in den vergangenen Jahren markant gestiegen. Viele Unternehmen versprechen sich dadurch eine Produktivitätssteigerung – auch Geschäftsanwendungen laufen vermehrt auf mobilen Geräten. Denn die Kunden wollen heute die Dienstleistung jederzeit und an jedem Ort beziehen: so auch die Kunden der SBB. Das neue Zugpersonalgerät, das seit Herbst 2005 eingesetzt wird, verkauft Billette wie am Schalter am Bahnhof und gibt Fahrplanauskünfte. Damit die Software auch auf der nächsten Hardware-Generation läuft, wurde sie plattformunabhängig programmiert.

Geschäftsanwendungen auf mobilen Endgeräten sind für Aussendienstmitarbeiter heute nahezu unverzichtbar. Neben Standardanwendungen auf Notebooks kommen immer mehr massgeschneiderte Endgeräte zum Einsatz. Dank flexibler

Peter K. Brandt

Kommunikation sowie leistungsfähiger Endgeräte lassen sich Lösungen realisieren, die noch vor wenigen Jahren undenkbar waren: Das Zugpersonalgerät für die SBB ist robust, berechnet einen Fahrplan innerhalb Sekundenbruchteilen und speichert die Tarifstruktur der ganzen Schweiz. Das Display lässt sich auch an der Sonne lesen, Barcode-Scanner, Drucker, GRPS, WLAN und Kartenleser sind in das Gerät integriert. Der Aussendienstmitarbeiter, bei der SBB der Zugbegleiter, ist damit in die IT-Landschaft des Unternehmens eingebunden. Informationen fließen ohne Medienbruch und ohne manuelle Bearbeitung zwischen allen Beteiligten. Im Feld erfasste Daten stehen in Echtzeit im Backend zur Verfügung und umgekehrt arbeitet das mobile Endgerät jederzeit mit aktuellen Informationen.

Software-Anwendungen wie das Zugpersonalgerät der SBB, die über viele Jahre auf einer grossen Zahl von Endgeräten eingesetzt werden, stellen grosse Herausforderungen an die Offenheit, Erweiterbarkeit und Portabilität. Da sich

im Lauf der Zeit ein Teil der Anforderungen ändert und neue hinzukommen, ist es essenziell, dass eine verbreitete Technologie genutzt wird, die auf offenen Standards basiert. Nur so kann sichergestellt werden, dass jederzeit Anpassungen und Erweiterungen durch die SBB selbst oder Dritte gemacht werden können. Von zentraler Bedeutung ist in diesem Kontext natürlich die Verfügbarkeit des kompletten Quellcodes, dessen Offenlegung heute bei Auftragsentwicklungen selbstverständlich sein sollte.

Doch nicht nur die Software ist Änderungen unterworfen. Auch die Hardware-Plattformen, auf denen die Client- und Server-Komponenten des Systems betrie-

ben werden, ändern sich über die Zeit. So muss mit Systemmigrationen im Backend gerechnet werden, bei denen neben neuer Hardware möglicherweise auch komplett andere Betriebssysteme zum Einsatz kommen. Noch wahrscheinlicher sind Änderungen beim mobilen Endgerät: Wenn eine Modellreihe eingestellt wird und keine Ersatzteile für die vorhandenen Geräte mehr lieferbar sind, ist oft eine Migration auf eine neue Umgebung nötig. Es sind Fälle bekannt, in denen eine aus Anwendersicht zufrieden stellende Lösung für neue Endgeräte komplett neu realisiert werden musste, da die bisherigen Geräte nach über zehnjähriger Nutzung starkem Verschleiss unterlagen und kein Ersatz verfügbar war.

Unabhängig von der Hardware

Gewünscht ist eine plattformunabhängige Software, die auf Rechnern mit unterschiedlichen Prozessoren, Systemarchitekturen und Betriebssystemen läuft. Dazu gibt es verschiedene Lösungen, wobei heute häufig das Konzept der «virtuellen Maschine» eingesetzt wird (siehe Kasten auf Seite 16). Dabei wird die Software für einen sehr detailliert definierten, aber nicht unbedingt real existierenden Computer formuliert. Dieser Computer, die virtuelle Maschine, wird wiederum in Form von Software auf die in Frage kommenden Plattformen aufgesetzt. Im Fall des neuen Zugpersonalgeräts der SBB, das unter dem Projektnamen ZPG II läuft, wurde die Java Virtual Machine (JVM) eingesetzt.

Mit einer plattformunabhängigen Software ist die Anwendung (beinahe) unabhängig von der konkreten Hardware. Dies hat zwei wesentliche Vorteile: Einerseits ist der Kunde unabhängig vom Hersteller der Hardware, da er die Software leicht portieren kann, andererseits wird bereits die initiale Hardware-Beschaffung erleichtert. So können die Informatiker an der Software arbeiten, während noch offen ist, auf welcher Hardware sie später laufen wird. Bei der Evaluation der Hardware liegt dann bereits ein Prototyp der zukünftigen Client-Software vor, mit der die angebotenen Geräte auf ihre Praxis-tauglichkeit überprüft werden können.

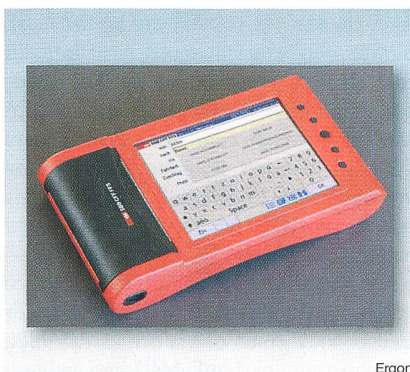


Bild 1 Das neue Zugpersonalgerät ZPG II wird seit Herbst 2005 bei der SBB eingesetzt.

Virtuelle Maschinen

Unter virtuellen Maschinen versteht man eine detaillierte Beschreibung eines typischerweise nicht real existierenden Prozessors sowie seiner Systemumgebung. Die Definition der virtuellen Maschine folgt zwei Zielen: Einerseits sollen

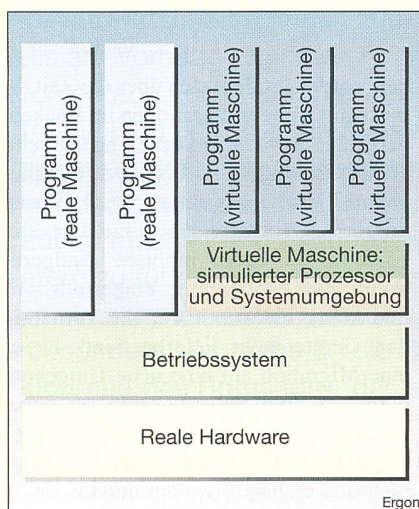


Bild 2 Konzept der virtuellen Maschine

leicht Programme für die beschriebene Umgebung aus einer Hochsprache erzeugt werden können (Anwendbarkeit). Andererseits muss es mit vertretbarem Aufwand möglich sein, eine Software-Implementierung der virtuellen Maschine für unterschiedliche Zielarchitekturen zu erstellen (Realisierbarkeit).

Der Nutzen von virtuellen Maschinen besteht darin, dass Programme, die dafür erstellt wurden, auf allen Plattformen laufen, für die eine Implementierung der virtuellen Maschine existiert. Zudem ist der «virtuelle Maschinencode» (oft Zwischencode oder Bytecode genannt) häufig so entworfen, dass er zusätzliche Informationen über das Quellprogramm enthält, was es ermöglicht, das Programm zur Laufzeit dynamisch zu optimieren. Moderne virtuelle Maschinen machen von dieser Möglichkeit Gebrauch. Dem Nutzen gegenüber steht der Zusatzaufwand, den entweder die Übersetzung oder die Interpretierung des Zwischencodes für die reale Plattform verursacht.

Das Konzept der virtuellen Maschine ist – wie vieles in der Informatik – keineswegs neu: Bereits UCSD-Pascal, eine Variante der Programmiersprache Pascal, die Ende der 60er-Jahre von Prof. Niklaus Wirth (ETH Zürich) entwickelt wurde, benutzte eine virtuelle Maschine namens P-System und den zugehörigen virtuellen Maschinencode P-Code, um die Realisierung von Pascal-Laufzeitumgebungen zu erleichtern (<http://www.threedee.com/jcm/psystem>).

Heute sind prominente Beispiele von virtuellen Maschinen die Java Virtual Machine (JVM), die von Sun Microsystems definiert wurde, sowie die Common Language Runtime (CLR) der .NET-Implementierung von Microsoft.

die Verfügbarkeit des Quellcodes wurde daher grosser Wert gelegt.

Die Funktionen des neuen Terminals sind im Vergleich zum Vorgänger stark erweitert worden. Neben der aus Passagiersicht zentralen Funktion des Billettverkaufs mit Bezahlung in Schweizer Franken, Fremdwährungen und mit Kreditkarte sind momentan folgende Anwendungen möglich:

- Fahrplanauskünfte (Papierfahrplan entfällt)
- Kontrolle elektronischer Tickets (siehe Bilder 3 bis 5)
 - OnlineTicket auf Papier
 - MobileTicket im Handy
 - RFID-Ticket auf kontaktloser Chipkarte (ähnlich der Oyster-Card in London oder der Suica-Card in Japan)
- Überprüfung von Abonnements (z.B. GA oder Monatsabonnement)
- Druck von Gutscheinen und anderen Spezialartikeln
- Kontrolle von Reisenden ohne gültigen Fahrausweis
- Nachschlagewerk für Zugsansagen (anstelle von Papier-Unterlagen)
- Erfassen betrieblicher Daten

Die Ausschreibung wurde in zwei Teilen vorgenommen: erst die Software und im Anschluss daran die mobile Hardware. Nach dem Zuschlag des Software-Teils an Ergon Informatik wurde bereits ein erster Prototyp der Software entwickelt, der in Feldtests zur Evaluation der verschiedenen mobilen Endgeräte diente. Die SBB entschied sich für das Gerät Almex.Mobile der Firma Höft&Wessel (siehe Tabelle). Erst nach der Entscheidung für die Hardware programmierten die Software-Entwickler die hardware-spezifischen Funktionen.

Die SBB ist bei der Ausschreibung des Projekts ZPG II – basierend auf früheren Erfahrungen – exakt so vorgegangen: Zuerst wurde die Software ausgeschrieben und erst später die Hardware. Dieses etappierte Vorgehen erlaubt einen deutlich späteren Hardware-Einkauf, was sich dank der steten Leistungszunahme vorteilhaft auf Einkaufspreise und Geräteleistung auswirkt.

Eine absolute Plattformunabhängigkeit erreichen selbst die fortgeschrittensten Techniken nicht. Bei spezialisierten Hardwarekomponenten wie Barcode-Scanner, RFID-Leseeinheiten oder selbst den eingebauten Druckern sind häufig keine hinreichenden Abstraktionen vorgesehen. Um hier keine unnötigen Abhängigkeiten einzuführen, empfiehlt es sich, die plattformspezifischen Eigenschaften hinter logischen Schnittstellen zu kapseln, damit bei einem späteren Plattformwechsel die

nötigen Anpassungen möglichst gering und eng lokalisiert sind (siehe dazu auch die Architekturbeschreibung des mobilen Clients weiter unten).

Das neue Zugpersonalgerät ZPG II

Wie der Name andeutet, war bei der SBB bereits vor dem neuen Zugpersonalgerät ein mobiles Terminal im Einsatz, mit dem im fahrenden Zug Billette verkauft werden konnten. Auf Grund verschiedener Einschränkungen und des Alters, das die früheren Geräte erreicht hatten, erfolgte im März 2003 die Ausschreibung für den Nachfolger.

Grundsätzlich war die SBB auf der Suche nach einer offenen und flexiblen Lösung, um im Lauf der Jahre weitere Anforderungen abdecken zu können. Auf die zukünftige Erweiterbarkeit und

Zugpersonalgerät ZPG II (Endgerät)	
CPU	Intel XScale mit 400 MHz
Betriebssystem	Windows CE.Net
Hauptspeicher	256 MB
Flash-Speicher	1024 MB
Peripheriegeräte	– Thermodrucker mit Papier-Barcodeleser – Chipkarten-Leser – Magnetkarten-Leser – RFID-Modul – 2D-Imager mit Kamerafunktion – CompactFlash Card Slot für externe Speicherkarte
Kommunikation	– GSM/GPRS – WLAN – Ethernet – USB – Infrarot-Schnittstelle

Technische Eigenschaften des mobilen Endgerätes ZPG II (Almex.Mobile von Höft&Wessel)

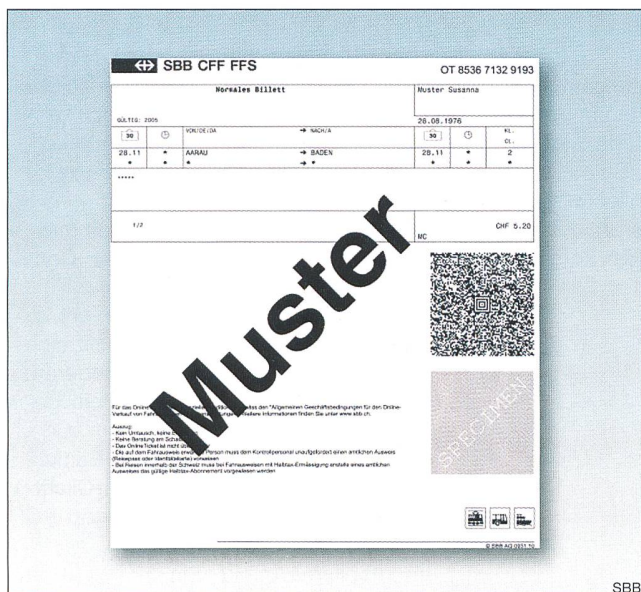


Bild 3 Das OnlineTicket der SBB wird vom Kunden selbst zu Hause ausgedruckt.



Quelle: SBB

Bild 4 Das Handy zeigt das MobileTicket an.

Gesamtarchitektur der realisierten Lösung

Die Java-Technologie gewährleistet die Plattformunabhängigkeit über alle Schichten. Die SBB hatte damit bereits in Vorgängerprojekten gute Erfahrungen gemacht. Zudem fügt sich Java bestens in die Gesamtstrategie ein, die auf die Offenheit der Systeme sowie Plattform- und Herstellerunabhängigkeit Wert legt.

Beim Backend auf dem Server kommt eine J2EE-Lösung (Java2 Enterprise Edition) zum Einsatz. Dem Administrator steht ein Java Rich Client (J2SE, Java2 Standard Edition) zur Verfügung. Die Anwendung auf dem mobilen Client für den Endnutzer schliesslich wurde als Applikation für J2ME (Java2 Micro Edition) in der Ausprägung CDC/PP implementiert. Einen Überblick über die verschie-

denen Java-Varianten gibt der Kasten auf Seite 18.

Die Gesamtarchitektur des ZPG-II-Systems zeigt Bild 7. Sie ist in mehrere Schichten unterteilt. In der Client-Schicht befinden sich die mobilen Endgeräte mit der Anwendung für den Einsatz im Zug. Ebenfalls zur Client-Schicht gehören die Administrationsanwendung und der Abrechnungs-Client. Die Clients greifen auf den J2EE-Server in der Application-Server-Schicht zu.

Der Server nutzt eine Oracle-Datenbank, um Daten persistent zu speichern. Schliesslich greift der Server auf Umsysteme zu, die sich in der Enterprise-Application-Integration-Schicht befinden. Als Application Server wird bei der SBB das Produkt IBM Websphere eingesetzt; bei der Datenbank handelt es sich um Oracle.

Der Application Server tauscht mit verschiedenen Umsystemen Daten aus. Via EJB (Enterprise Java Beans) wird mit den Multi Channel Services (MCS) der SBB kommuniziert, um auf verschiedene Live-Daten wie Kundendatenbank oder Online-Ticketing zuzugreifen. Andere Umsysteme sind über eine File-Schnittstelle angebunden, beispielsweise die Versorgung mit Sortimentsdaten oder die Ablieferung von Abrechnungs- und Verkaufsdaten sowie das Adressverzeichnis und andere Informationen, die das Zugpersonal in elektronischer Form statt auf Papier bei sich trägt.

Die mobilen Endgeräte kommunizieren über drei verschiedene Wege mit dem Application Server: Einerseits kann das ZPG II in einer Dockingstation platziert werden, die via das LAN eine Verbindung mit dem Server herstellt. Wenn sich das Mobilgerät im Bereich eines Public Wireless LAN Hotspots von Swisscom Mobile befindet, kann es darüber kommunizieren. In allen anderen Fällen wird über das Mobilfunknetz von Swisscom eine GPRS-Datenverbindung aufgebaut. Die Kommunikation zwischen Server und Client wird immer verschlüsselt, da die Verbindungen teilweise über öffentliche Netze gehen und die Vertraulichkeit der Daten essenziell ist.

Für alle drei Kommunikationsmedien ist ein spezieller Web Entry Server (WES; AirLock der Firma Seclutions) die Sicherheitsschranke am Eingang zum SBB Netzwerk. Der WES dient als zentraler Kommunikationsendpunkt für die verschlüsselte Verbindung vom mobilen Client. Die eintreffenden Anfragen werden mit verschiedenen Mechanismen auf potenzielle Angriffe auf den Application Server analysiert (Buffer Overflows, SQL injection attacks usw.). Gleichzeitig findet mit Hilfe des Firmen-Directorys eine zentrale Authentisierung und Rollen-Zuweisung statt. Für den Fall der GPRS-Kommunikation mit Mobilgeräten sind auf dem WES bestimmte Kommunikationsparameter gesetzt, um die Kommunikation mit TCP/IP über das Mobilfunknetz mit seinen spezifischen Eigen-



Bild 5 Die GraubündenCard nutzt RFID-Technologie.

Die Java-Familie

Java-Technologie wird in allen Bereichen eingesetzt – vom leistungsstarken Server über Desktop-PCs und mobile Geräte bis hin zu Chipkarten. Um den Anforderungen und Möglichkeiten in den verschiedenen Einsatzgebieten gerecht zu werden, wurden mehrere Varianten von Java definiert. Diese unterscheiden sich in den verfügbaren Programmierbibliotheken, aber auch in den Eigenschaften der virtuellen Maschine selbst. So stehen beispielsweise in manchen Umgebungen keine Fließpunktzahlen zur Verfügung oder es wird kein Garbage Collector vorgeschrieben.

Für die Verwendung als Application Server wurde J2EE (Enterprise Edition) definiert. Es stehen vielerlei Programmierschnittstellen für die Backend-Integration zur Verfügung. Zudem kommt das Konzept des Enterprise Application Containers zum Einsatz: In diesem können Server-Anwendungen installiert (deployed) und deinstalliert werden, ohne die

anderen Applikationen des Servers zu beeinflussen. Der Container stellt eine Umgebung bereit, die unter anderem auch die Transaktionssicherheit gewährleistet.

Die Variante J2SE (Standard Edition) wird für den Einsatz für Rich-Client-Applikationen auf dem Desktop-PC des Benutzers verwendet.

Im mobilen Umfeld von Interesse ist J2ME (Micro Edition). Da sich die Hardware bei mobilen Endgeräten stark unterscheidet, existieren verschiedene Konfigurationen, die durch so genannte Profile an den beabsichtigten Einsatz angepasst werden können. Die CLDC (Connected Limited Device Configuration) ist zusammen mit dem MID-Profil

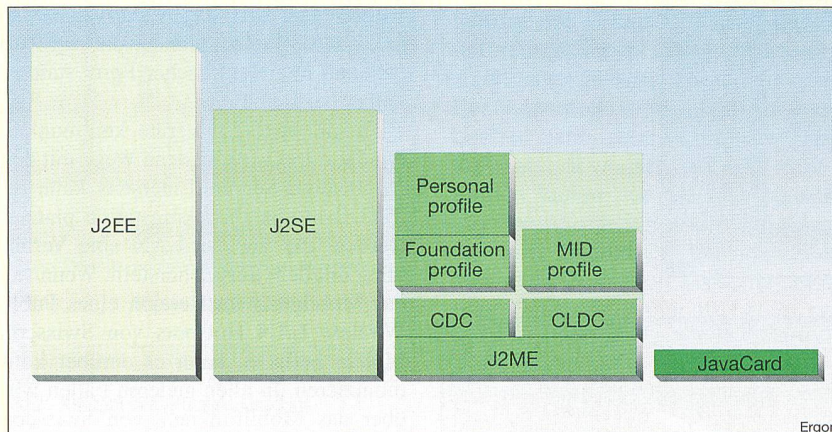


Bild 6 Die Java-Familie stellt je nach Variante mehr oder weniger Funktionen zur Verfügung.

(Mobile Information Device Profile) vor allem auf Mobiltelefonen anzutreffen. Auf leistungsfähigeren Geräten wie PDAs und Smartphones kommt in der Regel CDC (Connected Device Configuration) mit dem Foundation Profile oder dem Personal Profile zum Einsatz.

Den Abschluss bildet die Variante JavaCard, die speziell für die portable und sichere Applikationen für Chipkarten definiert wurde, da in diesem Umfeld noch wesentlich stärkere Einschränkungen und spezielle Anforderungen (z.B. Kryptografie) vorliegen.

schaften (hohe Latenz; geringe Bandbreite; Paketverluste) zu optimieren.

Bei der SBB ist es wesentlich, dass die Applikationssoftware und die Stammdaten auf den Mobilgeräten zentral verwaltet und über eine Kommunikationsverbindung aktualisiert werden können. Denn zusammen mit den Endgeräten von Partnerbahnen betreibt die SBB 2500 Mobilgeräte. Ein Rückruf der Geräte in eine zentrale Stelle für jedes Update würde einen enormen Aufwand bedeuten. Daher entschied sich die SBB schon früh dafür, Software und Stammdaten (Fahrpläne, Tariflisten, usw.) drahtlos über die vorhandenen Kommunikationsverbindungen zu aktualisieren. Dazu wurde in einer ersten Phase eine Erweiterungskomponente des WES für das applikationsspezifische Mobile Device Management (MDM) eingesetzt. Im Lauf des Projekts wurde diese durch eine generische MDM-Softwarelösung (IBM Websphere Everyplace Access) ersetzt, mit der bei der SBB zukünftig alle mobilen Anwendungen und Geräte verwaltet werden sollen.

Der mobile Client

Bild 8 zeigt den Aufbau des mobilen Clients. Neben der J2ME-Java-Umgebung kommt auf dem Client das von Ergon entwickelte Mobile Application Framework (MAF) zum Einsatz, das die gerätespezifischen Eigenschaften kapselt und abstrahiert. So werden beispielsweise Zugriffe auf Hardwarekomponenten wie 2D-Code-Scanner, Kartenleser oder Drucker durch eine geeignete Java-Schnittstelle gekapselt. Die Implementierung greift via «native Calls» auf die Bibliothek des Hardwareherstellers zu. Auf diesem Weg muss bei einer Portierung auf ein neues Gerät lediglich die Schnittstelle angepasst werden. Der Programmcode der Applikation ist dadurch nicht berührt.

Das Mobile Application Framework beinhaltet zudem Komponenten, die sich für mobile Softwarelösungen als hilfreich erwiesen haben. Neben spezialisierten Elementen für die grafische Benutzerschnittstelle sind dies unter anderem Hilfsmittel für die Kommunikation mit

dem Application Server und der Client-Teil des applikationsspezifischen Mobile Device Managements.

Bei der Anwendung wurde grosser Wert auf eine ergonomische Bedienung gelegt. Die Tastatur, die auf dem Bildschirm eingeblendet ist, wenn Eingaben erforderlich sind, kann der Durchschnittsanwender mit den Fingern bedienen. Bei der Eingabe in Freitextfelder wird möglichst früh eine Auswahl von Ergänzungsvorschlägen angezeigt, die mit einem Fingerdruck angewählt werden können.

Zudem wird – wann immer möglich – auf bereits eingegebene Daten zurückgegriffen. So zeigt das Gerät eine Liste der Verkäufe der aktuellen Schicht an, um diese Artikel erneut zu verkaufen. Auf diese Weise können Billette, die im gleichen Zug wiederholt nachgefragt werden (z.B. Klassenwechsel Zürich–Bern), effizient ausgestellt werden. Wo es sinnvoll ist, wird auch der im Gerät gespeicherte Fahrplan in Kombination mit der aktuellen Uhrzeit zu Hilfe genommen, um

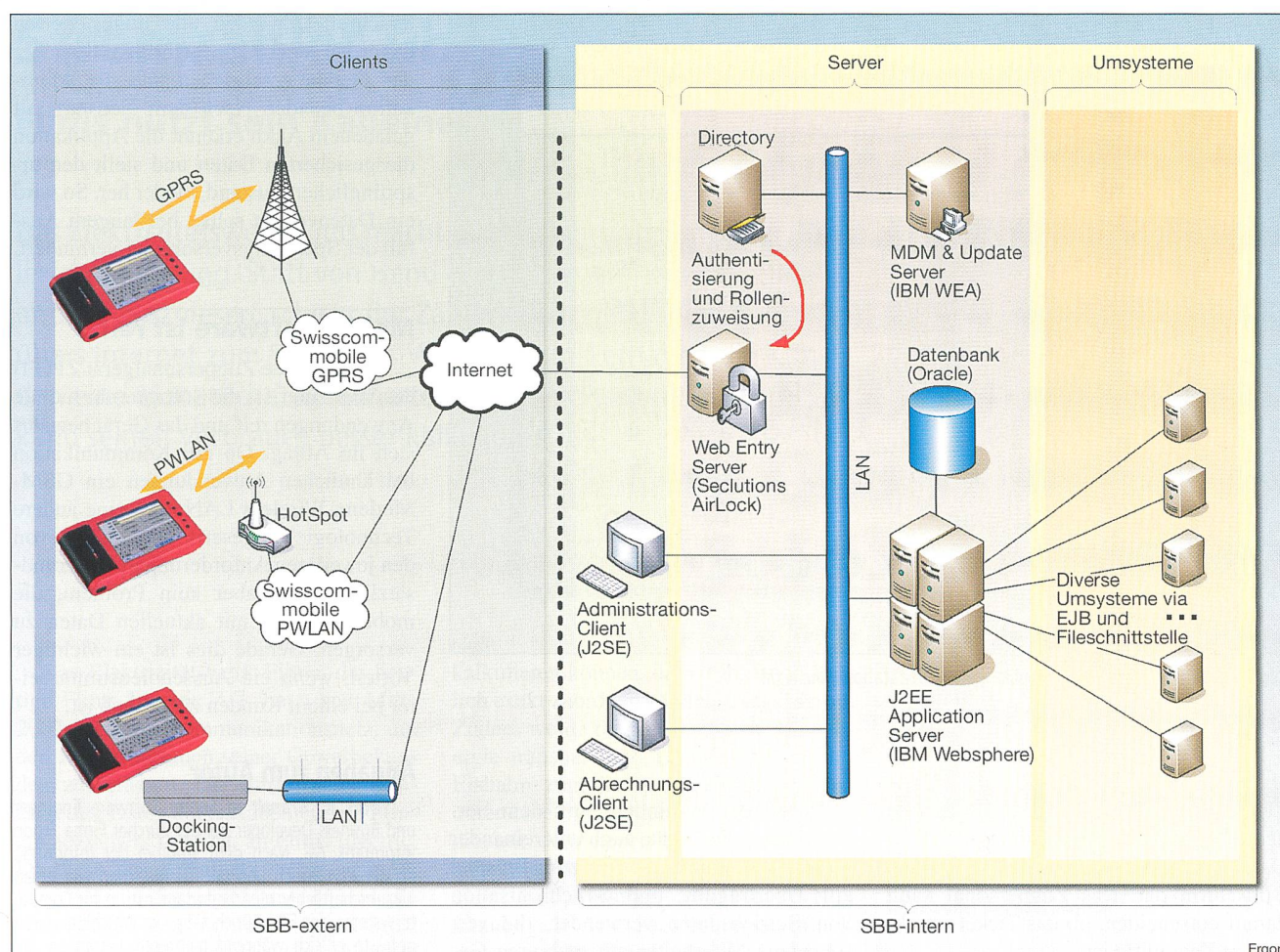


Bild 7 Gesamtarchitektur des ZPG-II-Systems

einen ersten Vorschlag für die momentane oder nächste Haltestelle zu machen.

Funktionen der mobilen Anwendung

Für den Verkauf im Zug ist es wichtig, dass die Billette zum selben Preis angeboten werden wie am Schalter oder am Billettautomat. Da die Algorithmen für die Preisberechnung komplex sind und sich regelmässig ändern, berechnet das Gerät die Preise nicht lokal. Eine Online-Abfrage beim Server für jedes Ticket kommt aber aus verschiedenen Gründen nicht in Frage: Wartezeit, Übertragungskosten, Funkloch im Tunnel usw. So müssen die vorausberechneten Preise gespeichert werden. Hierbei handelt es sich um eine grosse Datenmenge, die nicht permanent im Hauptspeicher der Anwendung gehalten werden kann. Daher wurde eine effizient gepackte und indizierte File-Datenstruktur entwickelt, mit der die Tarifdaten kompakt abgelegt und dennoch in kürzester Zeit ausgewertet werden können.

Da elektronische Tickets, die sich der Kunde selbst ausstellt, für die SBB immer

wichtiger werden, muss das Mobilgerät eine effiziente Kontrolle solcher Tickets erlauben. Derzeit sind zwei solche Ticketarten im Einsatz: das OnlineTicket, das sich der Kunde auf Papier ausdruckt, sowie das MobileTicket, das der Kunde per MMS auf sein Mobiltelefon zugestellt bekommt (Bilder 3 und 4). Zur Kontrolle

von OnlineTicket und MobileTicket beinhalten beide einen 2D-Code, der neben Name und Geburtsdatum des Fahrgasts auch die kompletten Ticketdaten (Startort, Zielort, Via, Klasse, Reisedatum usw.) enthält. Um Manipulationen auszuschliessen, ist die Information kryptografisch geschützt. Bei der Kontrolle im

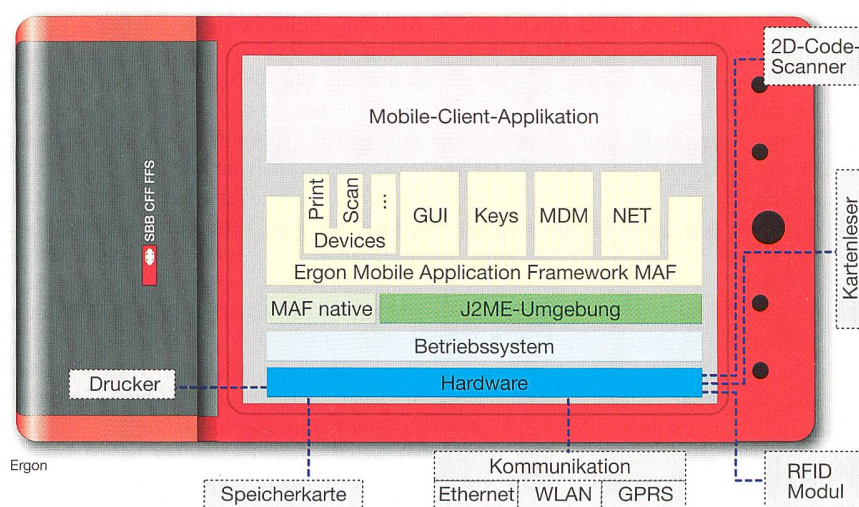


Bild 8 Aufbau des Zugpersonalgerätes ZPG II

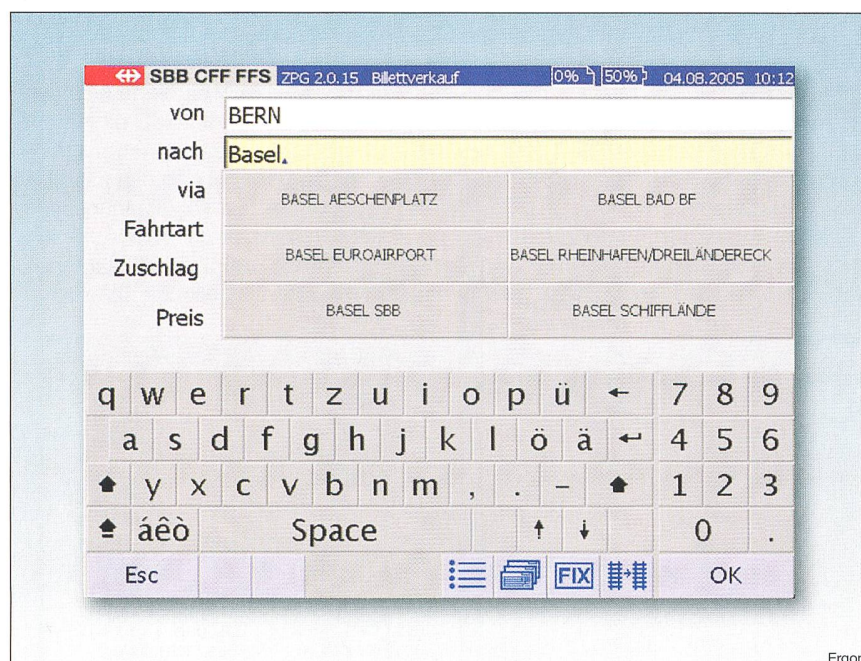


Bild 9 Screenshot der Verkaufssapplikation mit der Stationenauswahl

Zug erfasst ein Scanner des Mobilgeräts den 2D-Code, decodiert die enthaltenen Daten, überprüft sie und stellt sie auf dem Bildschirm dar. Das Zugpersonal kann darauf entscheiden, ob das Ticket im aktuellen Zug gültig ist.

Ähnlich wird die GraubündenCard kontrolliert. Dies ist ein im Kanton Graubünden eingesetztes elektronisches Ticket für den öffentlichen Verkehr: Eine kontaktlose Chipkarte, die auf RFID-Technologie basiert, ähnlich der Suica-Card von Japan Railways oder der Oyster-Card im Grossraum London (Bild 5). Damit die Mitarbeiter der Rhätischen Bahn, die ebenfalls mit dem ZPG II ausgestattet sind, den Inhalt der GraubündenCard eines Fahrgasts einsehen können, wurden diese Geräte mit einem RFID-Modul ausgestattet. Die Software des ZPG II liest damit die Daten von der Karte und zeigt sie dem Zugpersonal zur Kontrolle an.

Für die Kommunikation dienen im Feld zwei Medien. Falls in grösseren Bahnhöfen ein Public Wireless LAN HotSpot besteht, kommuniziert das Gerät darüber. Als Alternative steht praktisch flächendeckend GPRS über das Mobilfunknetz zur Verfügung. Aus Kosten- und Bandbreitenüberlegungen wird GPRS jedoch nur für spontane Echtzeit-Abfragen genutzt. Der regelmässige Daten-Upload bei Schichtabrechnung oder die Aktualisierung der Anwendung oder ihrer Stammdaten erfolgen über PWLAN, bzw. in manchen Fällen über eine Dockingstation.

Neben der Verbindung mit dem Server können die Geräte auch untereinander kommunizieren. Dies wird zum Beispiel zur Übertragung und Synchronisation von Betriebsdaten verwendet, die verschiedene Mitarbeiter mit mehreren Geräten vom aktuellen Zug erfasst haben. Dazu wird zwischen den Mobilgeräten über WLAN ein Peer-to-peer-Netzwerk aufgebaut.

Das Wichtigste bei mobilen Geräten mit unternehmenskritischen Anwendungen ist, dass sie robust und zuverlässig arbeiten. Die Anwendung soll permanent verfügbar sein und ein Datenverlust soll ausgeschlossen werden. Da von den Anwendern keine technischen Kenntnisse verlangt werden können, ist ein gewisses Mass an «Selbstheilung» wichtig.

Für das ZPG II der SBB sind verschiedene Massnahmen vorgesehen: Alle wesentlichen Daten, die unterwegs anfallen (z.B. Verkaufsdaten), werden auf zwei voneinander unabhängige Speichermedien geschrieben. Dabei verhindern kryptografische Verfahren unerwünschte Manipulationen. Zudem bedient der Benutzer nur die Anwendung, er hat keinen Zugriff auf das Betriebssystem. Dadurch werden mutwillige oder fahrlässige Fehlmanipulationen wie das Einschleppen eines Virus durch vermeintlich harmlose Zusatzprogramme verhindert.

Um einen Datenverlust bei Batterieausfall zu verhindern, wird der Ladezustand permanent überwacht. Sobald dieser einen definierten Schwellwert un-

terschreitet, werden alle ungesicherten Daten in den nichtflüchtigen Speicher des Geräts geschrieben und die Applikation beendet. Nach dem Neustart mit geladenem Akku erkennt die Applikation die gesicherten Daten und stellt den ursprünglichen Zustand wieder her. So wird ein Datenverlust selbst bei langem Ausfall der Spannungsversorgung verhindert.

Mobile Software ist reif

Wie das neue Zugpersonalgerät ZPG II der SBB zeigt, ist die Software für mobile Anwendungen reif und das Gerät bewährt sich im Alltag. Ob zur Kommunikation bei ähnlichen Anwendungen ein GSM-Modem, Wireless LAN oder eine andere Technologie eingesetzt wird, hängt von den jeweiligen Anforderungen ab. Grundsätzlich ist es aber kein Problem, die mobilen Geräte mit aktuellen Daten zu versorgen. Gerade dies ist ein wichtiger Vorteil, wenn ein Aussendienstmitarbeiter bei einem Kunden zu Besuch ist.

Angaben zum Autor

Peter K. Brandt ist Senior Software Engineer und Business Developer bei der Zürcher Firma Ergon Informatik AG. Nach dem Studium der Informatik an der ETH Zürich und der TU München war er als wissenschaftlicher Mitarbeiter am Institut für Computersysteme der ETH Zürich tätig. Im Anschluss daran befasste er sich während mehreren Jahren als Research Engineer bei Swisscom Innovations mit anwendungsbezogenen Forschungsthemen im Umfeld von Mobilität und Kommunikation. Seit 2001 ist Peter Brandt Mitarbeiter bei Ergon und dort für das Gebiet vernetzter mobiler Anwendungen auf Java-Basis verantwortlich.
Ergon Informatik AG, 8008 Zürich, www.ergon.ch

Résumé

Logiciel mobile indépendant du matériel

Appareil mobile pour le personnel de train des CFF. Le nombre d'appareils mobiles a fortement augmenté ces dernières années. De nombreuses entreprises en attendent une augmentation de la productivité – et de plus en plus d'applications commerciales tournent sur des appareils mobiles. En effet, les clients souhaitent actuellement obtenir les services partout et en tout temps: comme d'ailleurs les clients des CFF. Le nouvel appareil pour le personnel de train, en service depuis l'automne 2005, vend les mêmes billets qu'un guichet de gare et donne des renseignements d'horaire. Grâce à un logiciel approprié, les données sont toujours à disposition. Et afin que ce logiciel puisse tourner sur la prochaine génération d'appareils, il a été programmé indépendamment du matériel.