

Allgegenwärtiger Datenzugriff für mobile Geräte

Seminar *Smart Environments*, April 2004

Student: Marc Kramis, mkramis@student.ethz.ch

Betreuung: Juergen Bohn, bohn@inf.ethz.ch

Professor: Prof. Friedemann Mattern, mattern@inf.ethz.ch

Einführung

Die zunehmende Verbreitung von Computern und das wachsende Bedürfnis der Benutzer, Daten jederzeit, überall und ohne zusätzliche manuelle Intervention wie Backup oder Synchronisation bearbeiten zu können, verlangt nach neuen Lösungen.

Grundsätzlich lassen sich zwei verschiedene Ansätze unterscheiden: Der *persönliche Server* (Personal Server) und der *Allgegenwärtige Datenzugriff*. Beide profitieren von der rasanten technologischen Entwicklung und sind Gegenstand aktueller Forschung.

Der persönliche Server, ein Gerät in Sub-PDA-Grösse, erlaubt es einem Benutzer, seine Daten überall hin "mit zu nehmen" und mit Hilfe einer drahtlosen Verbindung auf lokal vorhandenen Computern zu bearbeiten oder Peripheriegeräte wie z.B. Drucker zu benutzen. Zusätzlich können persönliche Server mit anderen Geräten kommunizieren und Informationen über die Umgebung aggregieren oder Daten mit den persönlichen Servern anderer Personen austauschen.

Beim Allgegenwärtigen Datenzugriff hingegen wird eine Infrastruktur aufgebaut, welche die Daten unbemerkt zum Benutzer "hin transportiert". Die verwendete Middleware führt Aufgaben wie z.B. Daten-Synchronisation oder -Replikation und andere heute manuell getätigten Aktionen selbstständig durch und ermöglicht fließende Übergänge zwischen den Verbindungszuständen verbunden (connected) bzw. nicht verbunden (disconnected).

In der vorliegenden Arbeit werden beide Konzepte erläutert, verglichen, auf ihre Vor- und Nachteile hin untersucht sowie Forschungsschwerpunkte aufgezeigt. Es ist davon auszugehen, dass in Zukunft beide Ansätze koexistieren und sich als Hybrid sinnvoll ergänzen werden. Politisch dürfte die Problematik der Datenspuren in "smarten" Umgebungen im Zusammenhang mit der Privatsphäre noch einige Diskussionen provozieren.

1 Persönlicher Server

1.1 Konzept

Der persönliche Server (PS) ist ein kleines Gerät ohne direkte Ein- und Ausgabemöglichkeiten, das vom Benutzer überall hin mitgenommen werden kann und das mit der umliegenden Infrastruktur oder anderen persönlichen Servern über drahtlose Kommunikationsschnittstellen in Verbindung tritt. Der PS speichert alle privaten Informationen und Dokumente, welche auf gerade verfügbaren Computern oder anderen Geräten bearbeitet werden können. Zusätzlich können laufend Informationen über die Um-

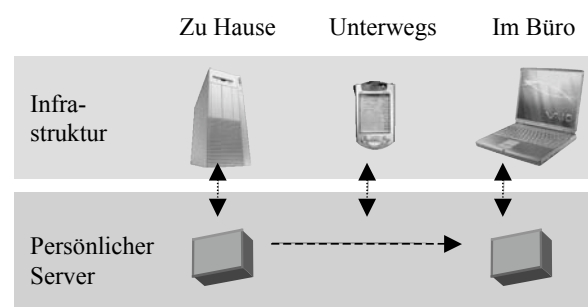


Bild 1: Konzept des persönlichen Servers

gebung zusammengetragen und abgespeichert werden.

Intel beispielsweise hat für die Entwicklung ihres Prototypen folgende Schwerpunkte gesetzt [2]:

- **Benutzer:** Einfache Benutzbarkeit, ohne mit den Schwierigkeiten eines verteilten, mobilen Systems konfrontiert zu werden.
- **Infrastruktur:** Einfache und schnelle Integration mit anderen Geräten in dynamischen drahtlosen Umgebungen.
- **Plattform:** Kleines Gerät mit genügend Speicherkapazität, Rechenleistung und Kommunikationsmöglichkeiten, um vom Benutzer überall hin mitgenommen werden zu können.

1.2 Implementation

Im Allgemeinen besteht die Hardware aus einem Rechenmodul, dem dazu passenden geräteinternen Ein- und Ausgabe Modul, einem oder mehreren Kommunikationsmodulen (z.B. Bluetooth, WiFi usw.), einem (nicht)flüchtigen Speicher für das Betriebssystem, einem Erweiterungsmodul für die Benutzerdaten und einem Akku für die Stromversorgung. Diese Komponenten werden in ein robustes und benutzerfreundliches Gehäuse verpackt, das resistent gegen Stöße, Staub und Spritzwasser ist.

Die Leistungsdaten des Prototyps der ersten Generation von Intel sind in Tabelle 1 aufgelistet [2]:

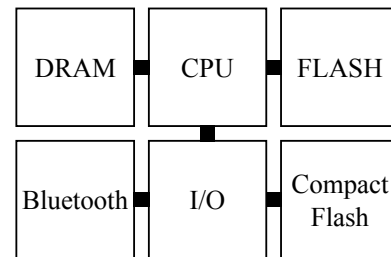


Bild 2: Hardware Architektur des Intel Personal Server [2]

Tabelle 1: Leistungsdaten des Intel Personal Server [2]

Modul	Typ	Info
Rechenmodul	StrongARM-1110	Taktfrequenz: 206MHz Leistungsaufnahme: <= 650mW
Ein- und Ausgabe Modul	StrongARM-1111	On-board USB
Speichermodule	FLASH DRAM	32MB 64MB
Erweiterungsmodul	Compact Flash	z.B. 1GB Microdrive (IBM)
Kommunikationsmodul	Bluetooth	Durchsatz: <= 723kbps
Akku	Li-Ion	Kapazität: 920mAh
Physische Eigenschaften	Daten	
Grösse (LxBxH)	Ca. 9x6x1.5cm	
Gewicht	Ca. 135g	
Leistungsaufnahme	1mW (Stand By) – 700mW (Vollbetrieb)	

Als Betriebssystem wird Linux verwendet. Physisch ist der PS mit dem Kommunikationspartner mittels Bluetooth Basisband verbunden. In der nächsten Generation kann die Verbindung wahlweise auch mit dem 802.11b (WLAN) Standard erfolgen. Auf der Netzwerkschicht wird eine IP Verbindung etabliert. Diese wiederum dient als Grundlage für diverse Dienstleistungen, die von Seiten des gerade durch den Benutzer verwendeten Gerät initiiert werden:

- **Administration und Fernzugriff:** Verwaltung des PS (z.B. Software Updates für den PS).
- **Internetdienst:** Erlaubt das Anzeigen von durch den PS bereitgestellten Webseiten.
- **Dateidienst:** Zugriff auf die Dateien des PS mittels WebDAV [10].

Damit erscheint der PS als vollwertiges System, auf das man standardisiert von verschiedenen Geräten aus zugreifen kann.

1.3 Vor- und Nachteile

Betrachtet man das Konzept des PS genauer, finden sich Vor- und Nachteile. Eine konkrete Implementation wie etwa die des Intel PS bringt zusätzliche Restriktionen mit sich. Eine Gegenüberstellung findet sich in Tabelle 2.

Tabelle 2: Vor- und Nachteile des PS

Konzeptionell		
++	Kontrolle	Der PS ist physisch für den Benutzer greifbar. Er bestimmt, wann und wo er den PS einsetzt. Zudem ist der Benutzer gewohnt, wichtige Gegenstände mit sich herum zu tragen.
++	Konnektivität	Der Benutzer ist nicht durch potentielle Netzwerkpartitionierungen behindert. Er kann jederzeit und überall auf seine Daten zugreifen.
+	Bandbreite	Der Benutzer kann jederzeit mit einer garantierten Bandbreite auf seine Daten zugreifen (je nach Kommunikationsschnittstelle sind andere Werte möglich).
+	Transparenter Datenzugriff	Der Datenzugriff erfolgt drahtlos und ohne manuelle Synchronisationsvorgänge oder Fernzugriffe.
+	Infrastruktur-Abhängigkeit	Der Benutzer ist unabhängig von einer bestimmten Hintergrund-Infrastruktur.
+	Komplexität	Die Hardware und Software bestehen aus standardisierten Komponenten.
-	Geräte-Abhängigkeit	Der PS muss lokal verfügbar sein, wenn man auf die Daten zugreifen will.
-	Teamwork / Datensynchronisation	Der PS bietet keine umfassende Unterstützung für Teamwork. Teamwork erfordert physische Präsenz der PS.
--	Datenverlust	Bei Verlust oder Diebstahl des PS kann es zu Datenverlust kommen.
Implementationsabhängig		
-	Bandbreite	Die Bandbreite bestimmter Kommunikationsschnittstellen ist für gewisse Anwendungen nicht ausreichend (insbesondere Bluetooth beim aktuellen Intel Prototypen).
-	Heterogenität der Schnittstellen	Viele Geräte bieten nur proprietäre Schnittstellen an. Der PS lässt sich daher nicht so einfach in bestehende Infrastrukturen einbetten.
-	Verfügbarkeit	Durch die begrenzte Akkulaufzeit wird die zeitliche Verfügbarkeit des PS eingeschränkt.

1.4 Anwendungsbeispiel

Als Anwendungsbeispiel des aktuellen PS von Intel sei die Verwendung des Dateidienstes beschrieben. Der auf dem PS unter Linux laufende Webserver ermöglicht Dateidienste mittels des WebDAV Protokolls [10], das auf HTTP aufbaut. In einem ersten Schritt erkennt ein Gerät automatisch den PS und bietet dem Benutzer in einem zweiten Schritt dessen öffentliche Verzeichnisse zur Verwendung an. Ab jetzt kann der Benutzer seine Dokumente bearbeiten, als ob sie lokal verfügbar wären. Die Kommunikation erfolgt somit über bekannte, standardisierte Schnittstellen.

1.5 Weiterentwicklung

Über das Prototypen-Stadium hinaus, das die Machbarkeit belegen sollte, geht es bei Intel nun darum, verschiedene Bereiche zu verbessern, um den PS fit für den alltäglichen Gebrauch zu machen [1,2]:

- **Drhtlose Kommunikation:** Einerseits soll die für das Auffinden von benachbarten Geräten benötigte Zeit von rund 21 Sekunden bei Bluetooth reduziert werden, um dem Benutzer eine noch direktere Interaktion zu ermöglichen. Andererseits ist die Datenrate von Bluetooth für gewisse Anwendungen mit effektiven 400kbps zu klein. Hier bietet sich der 802.11b WLAN Standard an, der parallel zu Bluetooth eingesetzt werden kann.
- **Energieverbrauch:** Wesentlich für die Akzeptanz ist die mögliche Betriebsdauer. Diese ist von der benötigten Rechen- und Kommunikationsleistung abhängig. Neue Verfahren für die Steuerung der Recheneinheit (z.B. Dynamic Voltage Management) und der Kommunikationsmodule (Effizienteres Auffinden von Geräten) können die Betriebsdauer verlängern. Bei den Akkus hingegen sind mittelfristig keine wesentlichen Verbesserungen zu erwarten.
- **Rechenleistung:** Damit der PS komplexere Aufgabenbereiche wie Spracherkennung oder Datenbanksysteme übernehmen kann, muss die Rechenleistung weiter gesteigert werden.
- **Integration:** Um den Benutzer noch weitergehend zu entlasten und den PS besser mit seiner Umgebung zu vernetzen, müssen alle Prozesse, die die Daten des PS betreffen, automatisiert ablaufen. Dazu zählt einerseits die Synchronisation zwischen dem PS und dem Gerät, das die Daten gerade bearbeitet, andererseits das Backup aller Daten, um einem Geräteverlust vorzubeugen.

Es ist anzumerken, dass die verschiedenen Verbesserungen gut abgestimmt werden müssen, weil sie sich zum Teil widersprechen. Als Beispiel seien die höhere Rechenleistung und der geringere Energieverbrauch erwähnt.

2 Allgegenwärtiger Datenzugriff

2.1 Konzept

Im Bereich des allgegenwärtigen Datenzugriffs (ADZ) gibt es eine ganze Reihe von Ansätzen, die unterschiedlich umfangreich und ausgereift sind. Hier soll eine Variante vorgestellt werden, die vielen Problemen gerecht wird und auch als entwicklungsfähige Basis herangezogen werden kann.

Als Grundlage dient die Idee, den ADZ mittels einer Middleware zu realisieren, die folgende drei Problemkreise angeht [6]:

- **Zugriff:** Der Benutzer will jederzeit und überall auf Daten zugreifen, ohne sich darum zu kümmern, ob Netzwerkverbindungen verfügbar sind bzw. ob er sich im Modus „verbunden“, „schwach verbunden“ (z.B. via GMS) oder „nicht verbunden“ befindet.
- **Geräteunabhängigkeit:** Der Benutzer will mit dem Gerät seiner Präferenz arbeiten und jederzeit das Gerät wechseln können, ohne manuell eine Synchronisation der Daten vornehmen zu müssen. Diese Geräte können in seinem privaten Besitz sein, Freunden oder auch der Allgemeinheit gehören (z.B. Computer an Hochschulen).

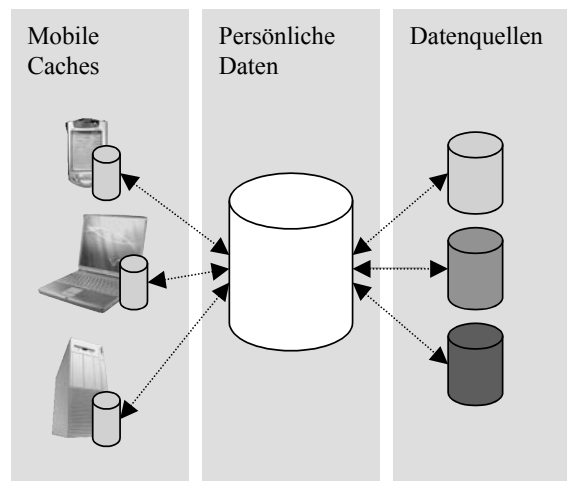


Bild 3: 3-Tier Architektur für allgegenwärtigen Datenzugriff [6]

- **Heterogene Datenquellen:** Der Benutzer will gleichzeitig auf unterschiedliche Datenquellen (Dateisysteme, Datenbanken, Mailserver, usw.) zugreifen können.

Die Infrastruktur wird dazu in drei Tiers aufgeteilt:

- **Datenquellen:** Diverse bestehende Datenquellen.
- **Persönliche Daten:** Zentrale Sammelstelle aller Daten eines Benutzers, realisiert als Data Warehouse
- **Mobile Caches:** Cache pro Gerät, der ein Subset der Persönlichen Daten enthält.

Durch die Einführung einer dritten mittleren Abstraktionsschicht (Persönliche Daten) erhält man die Möglichkeit, die mobilen Geräte von den Datenquellen zu trennen, um zentral alle Änderungen zu aggregieren. Diese können dann wiederum automatisch zu den mobilen Geräten oder Datenquellen propagiert werden, sobald sie mit dem Netzwerk verbunden sind. Das ermöglicht dem Benutzer eine einheitliche Sicht seiner Daten auf unterschiedlichen Geräten. Zusätzlich werden die Daten aus den heterogenen Datenquellen vereinheitlicht in den Persönlichen Daten abgelegt.

2.2 Implementation

Die Implementation des beschriebenen Konzepts wird detailliert in [6] beschrieben und soll hier nur zusammenfassend erwähnt werden. Die drei Tiers werden mittels eines regel- und ereignisbasierten Nachrichtensystems miteinander verbunden. Eine Menge von Regeln erzeugen Ereignisse die wiederum Transaktionen auslösen, die Daten zwischen den einzelnen Tiers hin- und her bewegen. Als Grundlage wird Linux und das Coda Dateisystem [8] verwendet. Einige interessante Aspekte dazu [6]:

- **Mobile Caches:** Solang das Gerät mit dem Netzwerk verbunden ist, wird direkt auf die externen Datenquellen zugegriffen. Jeder Zugriff wird als Ereignis abstrahiert und in Form von Metadaten gespeichert, die sukzessive zu den Persönlichen Daten propagiert werden, um diese konsistent zu halten. Bietet die Netzwerkverbindung nur eine geringe Bandbreite, werden die Informationen aus den Persönlichen Daten gemäss einer Prioritätenliste in den Mobilien Cache geladen. Bricht die Verbindung ganz zusammen, werden die Daten soweit möglich aus dem Cache geladen. Fehlende Daten werden nachgeladen, sobald wieder eine Verbindung steht.
- **Data Warehouse:** Es überwacht die definierten Datenquellen und synchronisiert diese mit den Persönlichen Daten. Zusätzlich werden Synchronisationsvorgänge bei eintreffenden Benutzerereignissen (Siehe Mobile Caches) ausgelöst. Dabei ist vorgesehen, dass der Benutzer nur bei komplexen aber seltenen Synchronisationsproblemen eingreifen muss. Normalerweise werden diese Vorgänge nach einem umfangreichen Regelwerk automatisch ausgeführt.
- **Inkrementelles Hoarding:** Damit nicht bei jedem Zugriff auf ein Dokument alle seine Daten kopiert werden müssen, was insbesondere bei kleinen Bandbreiten sehr zeitaufwändig ist, werden nur die geänderten Dokumentfragmente transferiert. Um dies zu realisieren, wird das Data Warehouse intern durch ein Versionierungssystem (Revision Control System) erweitert. Messungen haben ergeben, dass durch diese Massnahme lediglich noch 15-35% der Daten übertragen werden müssen. Der zusätzliche Speicheraufwand für das Versionierungssystem beträgt dabei rund 6-12%.

2.3 Vor- und Nachteile

Wieder soll zwischen konzeptionellen und implementationsabhängigen Vor- und Nachteilen unterscheiden werden. Die Gegenüberstellung findet sich in Tabelle 3.

Tabelle 3: Vor- und Nachteile des ADZ

Konzeptionell		
++	Datenverlust	Automatisierte Prozesse schützen vor Datenverlust.
++	Geräte-Abhängigkeit	Es kann frei, einheitlich und ohne zusätzlichen Aufwand auf eine Vielzahl von Geräten und Datenquellen zugegriffen werden.
+	Transparenter Datenzugriff	Der Datenzugriff erfolgt nach Verfügbarkeit drahtlos oder drahtgebunden und ohne manuelle Synchronisationsvorgänge oder Fernzugriffe.
+	Teamwork / Datensynchronisation	Der ADZ kann durch die bestehenden Synchronisations-Mechanismen einfacher als der PS für Teamwork-Szenarien angepasst werden. Zudem sind die Daten der Benutzer über das Netzwerk besser verfügbar als über die verschiedenen PS.
+/-	Konnektivität	Durch partitionierte Netzwerke kann es vorkommen, dass selten verwendete Daten bei Bedarf nicht bereit gestellt werden können.
+/-	Bandbreite	Die verfügbare Bandbreite kann stark variieren.
-	Kontrolle	Dem Benutzer wird die direkte Kontrolle über seine Daten weitgehend entzogen. Er muss sich auf die Infrastruktur verlassen können. Diese speichert und transferiert die Daten.
-	Komplexität	Das Zusammenspiel der verschiedenen Komponenten ist nicht trivial, insbesondere wegen der variablen Konnektivität und der Komplexität der Definition des Regelwerkes.
--	Infrastruktur-Abhängigkeit	Der ADZ benötigt zwingend eine Infrastruktur, die zuerst aufgebaut werden muss (Softwareinstallation).
Implementationsabhängig		
-	Heterogenität der Geräte und Schnittstellen	Viele Geräte bieten nur proprietäre Schnittstellen an. Die Middleware wird vorerst nur für eine begrenzte Zahl von Geräten und Datenquellen verfügbar sein. Zudem sind verschiedene Varianten des ADZ nicht a priori kompatibel zueinander

2.4 Anwendungsbeispiel

Der Benutzer bearbeitet seine Dokumente zu Hause und im Geschäft. An beiden Orten sind die aktuellen Dokument-Versionen lokal im Mobile Cache verfügbar. Die Middleware des ADZ nimmt automatisch die Daten-Synchronisation zwischen den beiden Mobile Caches und den zentralen Persönlichen Daten vor. Weiter kopiert sie im Rahmen des inkrementellen Hoarding Dokumente bzw. Dokumentfragmente aufgrund der Zugriffsstatistik von den Persönlichen Daten wieder zu den betroffenen Mobile Caches. Konkret findet der Benutzer jederzeit und überall die aktuelle Version seiner Dokumente vor, ohne dass er manuell Daten kopieren muss.

2.5 Weiterentwicklung

Mit ersten Prototypen konnten umfangreiche Tests durchgeführt werden, die den Machbarkeitsbeweis lieferten [6]. Dennoch ist im gesamten Spektrum von Dateisystemen über Data Warehouses bis zu den Mobile Caches eine umfangreiche Weiterentwicklung nötig, um den ADZ zur Marktreife zu bringen. Dazu entwickeln verschiedene Gruppen bestehende Konzepte weiter [6]:

- **Betriebssysteme:** Die Middleware muss für alle gängigen Betriebssysteme verfügbar sein, was nicht zu letzt beim Marktführer Windows eine grosse Hürde darstellt, weil die Entwicklung eines integrierten Dateisystems ohne Kenntnis der internen Betriebssystemeigenschaften schwierig ist.
- **Data Warehouse:** Das Data Warehouse ist eine kritische Komponente im System, die verschiedenen Anforderungen genügen muss. Die Forschung konzentriert sich in diesem Bereich auf die automatische Daten-Synchronisation und -Replikation, um dem Data Warehouse einen weitgehend autonomen Betrieb zu ermöglichen [3].
- **Darstellung und Formate:** Die Integration von verschiedenen Datenquellen, unterschiedlichen Darstellungsfähigkeiten einzelner Geräte (Bildschirmgrösse, Auflösung, Farbtiefe) und variable Bandbreiten für die Übertragung der Daten erschweren den Einsatz von einheitlichen Formaten oder Darstellungsarten. Jedes Format bringt gewisse Vor- und Nachteile mit sich. Qualität und Funktionsvielfalt korrelieren meist mit dem Speicherplatzbedarf. Eine problemlose Konvertierung ist nicht immer trivial, weil viele Anwendungs-Programme Inhalt und Form nicht sauber trennen und solche Operationen meist nicht invers zueinander sind. In vielen Fällen ist die Konvertierung nicht möglich.

3 Vergleich

Vergleicht man die Konzepte des PS und des ADZ aufgrund ihrer Vor- und Nachteile, ergibt sich folgendes Bild in Tabelle 4:

Tabelle 4: Vergleich der Konzepte PS und ADZ

Bereich	PS	ADZ
Bandbreite	+	+/-
Datenverlust	--	++
Geräteabhängigkeit	-	++
Infrastrukturabhängigkeit	+	--
Komplexität	+	-
Konnektivität	++	+/-
Kontrolle	++	-
Teamwork / Datensynchronisation	-	+
Transparenter Datenzugriff	+	+

Als wesentliche Pluspunkte für den PS seien Kontrolle und Konnektivität genannt. Diese geben dem Benutzer nicht zuletzt auch psychologisch eine grössere Sicherheit. Der ADZ punktet beim Schutz vor Datenverlust und der Tatsache, dass der Benutzer nicht an ein bestimmtes Gerät (wie den PS) gebunden ist. Insgesamt weist das Konzept des PS genau dort seine Schwächen auf, wo der ADZ seine Stärken hat und umgekehrt.

Bei der Implementationsreife fällt auf, dass der PS wesentlich weiter entwickelt ist, als der ADZ. Dies hängt mit der Tatsache zusammen, dass der PS eine begrenzte und überschaubare Welt erschafft, die zum grössten Teil bestehende, standardisierte Komponenten integriert. Wesentlich ist die Fähigkeit, über bekannte Schnittstellen zu kommunizieren. Dies führt zu einer einfacheren Verwendbarkeit und einer reduzierten Komplexität. Zudem folgt der PS einem wichtigen Konzept der Informatik, dem Information Hiding. Er bietet nach aussen wohldefinierte Schnittstellen, kann aber intern als Black Box betrachtet werden. Der ADZ verwendet im Gegensatz dazu viele interne Konzepte und Ideen, die noch nicht Marktreife erreicht haben und in ihrer Definition und Semantik wesentlich komplexer sind (z.B. Regeln und Ereignisse) und einige Erfahrung verlangen. Das Reibungslose Zusammenspiel bzw. die Stabilität des Systems ist noch nicht vollumfänglich ge-

währleistet. Die Forderung nach einer einheitlichen globalen Infrastruktur lässt sich daher nicht so schnell umsetzen – wenn überhaupt.

3.1 Frühe Anwendung

Während der ADZ heute nur in streng begrenzten Forschungsumgebungen funktioniert, gibt es bereits erste frühe Vorläufer des PS im täglichen Einsatz. Diese sind in ihrer Funktionalität stark eingeschränkt oder für zusätzliche Anwendungsbereiche ausgelegt, erfreuen sich aber bei den Benutzern grosser Beliebtheit. Die Rede ist von USB Memory Sticks oder von Geräten wie dem iPod (Funktion als Datenspeicher). Diese erlauben aber nur einen drahtgebundenen Zugriff auf Dateien und verfügen über keine oder nur eingeschränkte bzw. spezifische eigene Rechen- und Dienstleistungsmöglichkeiten im Sinn eines PS. Trotzdem zeigen sie das Potenzial und die Akzeptanz des Konzepts des PS sehr gut auf.

3.2 Künftige Anwendung

Aufgrund der Möglichkeiten und Schwächen beider Ansätze ist davon auszugehen, dass sie jeweils nach Anforderung in verschiedenen Anwendungsbereichen (parallel) eingesetzt werden. Der ADZ dürfte sich vorwiegend in geschlossenen Systemen von Organisationen durchsetzen, weil sich die Infrastruktur innerhalb deren Grenzen sehr gut kontrollieren und verwalten lässt. Der PS hingegen kann seine Trümpfe bei der Integration heterogener Infrastrukturen (z.B. für ADZ) ausspielen, weil er allgemein anerkannte Schnittstellen bietet.

Beide Konzepte werden gleichermaßen von Entwicklungen und insbesondere Standards im Bereich der Daten-Synchronisation und –Replikation profitieren. Gerade beim PS ist das Backup bzw. die Replikation entscheidend, um sich vor Verlust des Gerätes schützen zu können.

Beim PS stellt die physische Kombination mit anderen Geräten eine weitere Einsatzmöglichkeit dar. Wie bereits Handy und PDA zum Smartphone verschmolzen wurden, könnte das Smartphone mit dem PS zu einer Art „Persönlichem Smartphone Server“ kombiniert werden. Andere Varianten der Kombination sind mit den smarten Kleidern oder sogar Computern als Schmuck bereits angedacht. Der PS könnte so langfristig zu einem ständigen Begleiter des Menschen werden.

Beim ADZ ist es wichtig, dass umfassende und von allen akzeptierte Standards geschaffen werden, damit eine einheitliche globale Infrastruktur überhaupt entstehen kann. Gerade diese Forderung macht angesichts der Erfahrungen mit diesem Themengebiet im Sektor der Informatik skeptisch. Es dürfte nicht realistisch sein, ganze monolithische ADZ Konzepte zu standardisieren, sondern viel mehr müssen kleine Module gebildet werden, die bereits für sich genommen einen genügend grossen Wiederverwendungswert ausserhalb des ursprünglichen Einsatzgebietes des ADZ haben.

Langfristig dürften sich hybride Ansätze durchsetzen, die von den Vorteilen beider Konzepte gleichermaßen profitieren und dementsprechend die Nachteile kompensieren sollen. Gerade im Einsatz als replizierter mittlerer Tier oder Mobile Cache im ADZ könnte der PS einen wesentlichen Beitrag leisten.

4 Datenspuren und Privatsphäre

Bei den beiden vorgestellten Konzepten tritt das Phänomen auf, dass Daten von Benutzern ausserhalb ihrer angestammten privaten Umgebung anfallen. Damit direkt Verbunden sind die zugehörigen Metadaten (Zeit, Ort, Benutzer, Dokumente usw.), deren Management und Auswertung sich der direkten Kontrolle des Einzelnen entziehen. Es entsteht eine Art von Datenspuren in smarten Umgebungen.

Wenn Benutzer ihre Dokumente auf fremden oder öffentlichen Rechnern bearbeiten, gibt es keine Garantie, dass diese Daten wieder gelöscht werden und nicht in Form von Sicherheitskopien, temporären Dokumenten usw. zurückbleiben. Im Moment, wo das Dokument bearbeitet wird, können potentiell sensible persönliche Daten eines Benutzers preisgegeben werden. So sind selbst verschlüsselt gespeicherte Daten via Bildschirm durch dritte einsehbar oder lassen sich via spezielle Programme direkt vom Gerät auslesen.

Die Infrastruktur kann jeden Kontakt mit einem PS zeitlich und örtlich lokalisieren. Es ist keinesfalls klar, ob der PS anonym oder mit einer offenen eigenen Identität versehen ist. Somit können im einfachsten Fall allgemeine Statistiken erstellt und Verhaltensmusteranalysen durchgeführt werden. Im extremsten Fall lassen sich Benutzerprofile erstellen, die über das Leben eines Menschen detailliert Auskunft geben können. Beim ADZ werden diese Informationen sogar aktiv benötigt, um etwa Hoarding durchzuführen.

Damit der Schutz der Privatsphäre des einzelnen Benutzers gewährleistet ist, muss eindeutig festgelegt werden, wie Daten und Metadaten zu verwalten sind und wer auf diese Zugriff hat. Ein Ansatz dazu bietet das Platform for Privacy Preferences Project (P3P) des W3C [9]. Ein Standard wäre wünschenswert, nach dem die einzelnen betroffenen Komponenten zertifiziert werden könnten. Zu diesen Komponenten gehören z.B. das Betriebssystem, Infrastruktur-Middleware, Dokumentbearbeitungssoftware oder PS. Diese Aufzählung lässt erahnen, welche Komplexität und politische Dimension sich aus dieser Anforderung nach einer klaren und verbindlichen Regelung der Privatsphäre ergibt. In diesem Bereich sind noch viele Diskussionen zu erwarten. Es ist zu hoffen, dass eine benutzerfreundliche und praktikable Lösung gefunden wird, die die Privatsphäre des Individuums angemessen respektiert.

Zusammenfassung

In der vorliegenden Arbeit wurden zwei Konzepte vorgestellt, die dem Benutzer den allgegenwärtigen und unkomplizierten Zugriff auf seine Informationen ermöglichen sollen. Dabei wird der verbreitete Einsatz von persönlichen Servern bereits mittelfristig als realistisch beurteilt. Der allgegenwärtige Datenzugriff hingegen dürfte aufgrund seiner Komplexität noch einige Zeit auf sich warten lassen und sich vor allem in geschlossenen Umgebungen durchsetzen. Langfristig dürften beide Konzepte koexistieren und im hybriden Einsatz neue Möglichkeiten eröffnen.

Literaturverzeichnis

- [1] INTEL: *Intel® Personal Server Research: Mobile Computing In The Palm of Your Hand*. Intel Research, 2004.
- [2] Roy Want, Trevor Pering, Gunner Danneels, Muthu Kumar, Murali Sundar, and John Light: *The Personal Server: Changing the Way We Think About Ubiquitous Computing*. Intel Research, June 2002.
- [3] R. Want, T. Pering, D. Tennenhouse: *Comparing autonomic and proactive computing*. IBM SYSTEMS JOURNAL, VOL 42, NO 1, 2003.
- [4] Abdelsalam (Sumi) Helal and Joachim Hammer: *UbiData: Ubiquitous Data Access for Mobile Users*. Submitted to the ACM SIG-Record, March 2003.
- [5] Jinsuo Zhang, Abdelsalam (Sumi) Helal and Joachim Hammer: *UbiData: Ubiquitous Mobile File Service*. Proceedings of the ACM Symposium on Applied Computing (SAC), Melbourne, Florida, March 2003.
- [6] Sumi Helal, Joachim Hammer, Jinsuo Zhang, and Abhinav Khushraj: *A Three-tier Architecture for Ubiquitous Data Access*. Proceedings of the First ACS/IEEE International Conference on Computer Systems and Applications, to be held June 2001.
- [7] Friedemann Mattern: *Ubiquitous Computing: Szenarien einer informatisierten Welt*. ETH Zürich, 2004.
- [8] Satyanarayanan, M.: *Coda: A Highly Available File System for a Distributed Workstation Environment*. Proceedings of the Second IEEE Workshop on Workstation Operating Systems, Sep. 1989, Pacific Grove, CA.
- [9] Website P3P of the W3C: <http://www.w3.org/P3P>, April 2004.
- [10] Website WebDAV: <http://www.webdav.org>, April 2004.