

Signalisierungsplattform zur netzübergreifenden Dienststeuerung in heterogener TIME Infrastruktur

Wolfgang Kellerer, Peter Sties

Technische Universität München (TUM)
Lehrstuhl für Kommunikationsnetze (LKN)
Arcisstraße 21, 80333 München, Germany
{kellerer, sties}@ei.tum.de, <http://www.ei.tum.de/>

Zusammenfassung. Die heutige Kommunikationsinfrastruktur ist geprägt von einer wachsenden Heterogenität von Fest- und Mobilfunknetzen, sowie einer Vielzahl unterschiedlicher Endgeräte aus den Bereichen Telekommunikation, Information und Medien / Entertainment (TIME). Eine weitreichende Nutzbarkeit von Informations- und Kommunikationsdiensten setzt daher einen Zugang zu diesen Diensten unabhängig von der Netzinfrastruktur voraus. Darüberhinaus müssen diese Dienste über bislang getrennte Netze übergreifend arbeiten können. Um eine derartige Konvergenz heterogener Netze auf Dienstebene zu erreichen, wurde am Lehrstuhl für Kommunikationsnetze der Technischen Universität München eine Signalisierungsplattform für die netzunabhängige Dienststeuerung entwickelt. Unsere Architektur beschreibt ein komponentenorientiertes Serversystem, das übergeordnet zu den Netzen angeordnet ist.

In der vorliegenden Arbeit geben wir einen Überblick über das Gesamtkonzept und beschreiben die Komponenten der Plattform, die die Netzunabhängigkeit realisieren: Die Netzadaptoeren. Neben Netzadaptoeren, welche die Architekturinterne Signalisierung auf Netz-API umsetzen, gibt es spezielle Adaptoeren für den Teilnehmerzugang, für den Zugang zu Content Ressourcen und für Spezialressourcen. Ein Lookup-Server zur dynamischen Registrierung der Adaptoeren vervollständigt die Architektur. Die Kommunikation zwischen den verteilten Komponenten der Architektur erfolgt über ein Signalisierungsprotokoll, das in Anlehnung an das Session Initiation Protocol (SIP) der IETF definiert wurde.

1 Einführung

Die derzeitige Kommunikationslandschaft ist geprägt von verschiedenen Kommunikationssystemen, die als verteilte Systeme unterschiedliche Anforderungen erfüllen. Gerade in der heutigen Zeit, die von der Konvergenz bisher getrennter Informations-, Kommunikations- und Verteildienste geprägt ist, reicht ein System alleine nicht aus, um alle Teilnehmeranforderungen (Qualität, Bandbreite, Dienstvielfalt) zu erfüllen. Vielmehr sind an der Erbringung von Teilnehmerdiensten viele Instanzen in unterschiedlichen Protokollebenen und an verschiedenen Stellen in den Netzen (z.B. Server, Netz, Endgerät) beteiligt. Eine Vielzahl konkurrierender oder sich ergänzender Kommunikationssysteme bildet eine stark heterogene Landschaft aus Standards und Systemen in den TIME Bereichen Telekommunikation (z.B. ISDN, GSM), Information (z.B. Internet, WAP) und Medien / Entertainment (z.B. CATV, DVB). In allen Berei-

chen ist neben den Bestrebungen zur Vereinheitlichung z.B. mit B-ISDN/ATM oder UMTS/IMT2000 derzeit eher eine starke Tendenz zur weiteren Diversifikation zu beobachten. Bestehende Systeme werden weitergenutzt, um den steigenden Bandbreitenbedarf zu decken, bzw. bisher unbeachtete Systeme erfahren einen neuen Nutzen (z.B. Data Broadcasting, Powerline Communication). Daher ist es auch in Zukunft äußerst unwahrscheinlich, daß eine einzige, einheitliche Netzinfrastruktur vorherrschen wird.

Der laufende Wettbewerb unter den Anbietern von Informations- und Kommunikationsdiensten für die Endteilnehmer, der durch die jüngste Deregulierung auf dem Telekommunikations- und Informationsmarkt noch weiter verschärft wird, trägt darüberhinaus dazu bei, daß der Bedarf wächst, verschiedene Netze zu kombinieren, um auf diese Weise mehr Teilnehmer ansprechen zu können. Es sind die Dienste, die die verschiedenen Anbieter untereinander auszeichnen. Diese Dienste müssen schnell über unterschiedliche Netze an die Teilnehmer gebracht werden, um einen möglichst großen Marktanteil zu erzielen. Die Einbeziehung neuer, bisher anders genutzter Netze, wie z.B. Verteilnetze für Kommunikationsdienste, bringt dabei weitere Vorteile, um sich von der Konkurrenz abzuheben und eine größere Zahl von Teilnehmern zu erreichen.

Dienste sind einem schnellen Lebenszyklus unterworfen. Um die zügige Einrichtung und Erbringung von Diensten auf unterschiedlichen Netzen zu ermöglichen, sind neue Lösungen gefordert, die eine netzunabhängige Beschreibung und Steuerung der Dienste unterstützen und zusätzlich von den verschiedenen Endgeräten der Teilnehmer abstrahieren.

Um diesen Anforderungen Rechnung zu tragen wurde am Lehrstuhl für Kommunikationsnetze der Technischen Universität München eine Architektur für eine netzunabhängige Dienststeuerung entwickelt, die eine Steuerung von Diensten auf heterogenen Netzen realisiert. Dabei sind folgende Anforderungen berücksichtigt:

- Heterogenität der Netze und unterschiedliche Endgeräte für die Dienststeuerung transparent machen
- Unabhängigkeit der Dienststeuerung von netzspezifischen Steuerungsdetails
- Ausnutzung von vorhandenen netzspezifischen Diensten über bestehende APIs
- Kombination von Netzen zur übergreifenden Dienststeuerung
- Schnelle und einfache Dienstentwicklung und Erweiterbarkeit
- Adaptivität bezüglich sich ändernder Infrastruktureigenschaften
- Einfache Zugangsmöglichkeit für Teilnehmer über unterschiedliche Netze/Kommunikationstechnologien

In der vorliegenden Arbeit werden nach einem Überblick über das Gesamtkonzept die Mechanismen unserer Plattform beschrieben, welche die Unabhängigkeit der Steuerung von den spezifischen Eigenschaften und den eigenen Signalisierungsprotokollen der Netze realisieren. Das ist neben den unterschiedlichen Adaptorkomponenten für Teilnehmerzugang und Netzsteuerung insbesondere das neu entwickelte

Signalisierungsprotokoll *SesCP*, das die asynchrone Kommunikation zwischen den Steuerungskomponenten und den Adaptor-Komponenten sicherstellt.

Die weitere Arbeit gliedert sich wie folgt. Zunächst verdeutlichen in Abschnitt 2 einige Dienstszenarien das Ziel unseres Ansatzes. Ein Überblick über die netzunabhängige Dienststeuerplattform und die grundlegenden Konzepte wird in Abschnitt 3 gegeben. Abschnitt 4 geht näher auf die Netz-Adaptor-Komponenten ein, die die Netz-unabhängigkeit realisieren. Das Signalisierungsprotokoll, über das die verteilten Komponenten der Architektur kommunizieren wird anschließend in Abschnitt 5 beschrieben. Zusammenfassung und Ausblick schließen die Arbeit ab.

2 Dienste und Dienstszenarien

Die Dienststeuerplattform ist zu den heterogenen Netzen und den Netzen unterschiedlicher Betreiber übergeordnet angesiedelt. Sie stellt allen Teilnehmern in den angeschlossenen Netzen ihre Dienste bereit. Es ist insbesondere Aufgabe der Dienststeuerung *höhere Dienste* zu erbringen, die über die Funktionalität der einzelnen Netze hinausgehen. Dies umfaßt zum einen die Funktionalität, Dienste übergreifend in heterogenen Netzen zu steuern. Zugangnetz und Ausführungsnetz können dabei vollständig getrennt verwaltet werden. Zum anderen ermöglicht die Plattform eine Kombination von bisher getrennten Kommunikationsdiensten, um auf diese Weise neue Dienste zu schaffen (z.B. Konferenz plus Video Abruf). Darüberhinaus erlaubt die Programmierbarkeit der Dienststeuerung die Entwicklung neuer, teilnehmerspezifischer Dienste.

Die folgenden Beispiele zeigen mögliche Szenarien zukünftiger netztransparenter Dienstnutzung.

- Während einer Telefonkonferenz zwischen einem H.323 Multimedia PC und einem GSM Telefon beschließen die Teilnehmer eine Multimediapräsentation aus dem Internet zu Rate zu ziehen. Während der H.323 Teilnehmer diese direkt empfangen kann, erhält der mobile Teilnehmer die Präsentation auf seinem ebenfalls mitgeführten Laptop über ein terrestrisches digitales Rundfunknetz.
- Ein Unternehmensberater mit wechselnden Aufenthaltsorten greift auf seine Intranet-Informationen-Dienste über Internet-Browser, WAP-Handy oder per Sprachzugang in seinem Dienstwagen zu.
- Während einer ISDN basierten Multimedia-Konferenz wird ein Teilnehmer über das Internet und ein weiterer über UMTS zugeschaltet, ohne daß die Netze explizit von den Teilnehmern angewählt werden.
- Über seinen zentralen Kalender kann ein Geschäftsmann seine globale Erreichbarkeit automatisch konfigurieren. Dabei kann er nicht nur seine persönlichen Einstellungen über unterschiedliche Netze/Dienste vornehmen (WWW, Email, Voice), sondern diese gelten auch für alle Kommunikationssysteme, die er benutzt.

3 Netzunabhängige Signalisierungsplattform

Um die oben beschriebenen und illustrierten Anforderungen zu erfüllen, wurde eine neuartige Dienstarchitektur definiert, die sich insbesondere durch eine Strukturierung in funktionale Module auszeichnet. Bild 1, rechts, zeigt das zugrundeliegende Referenzmodell, in dem die unabhängigen Funktionen unterschieden werden.

Anwendungen (Applications) werden hier von der reinen Dienststeuerung (Service Control) getrennt. Die Dienststeuerung verwaltet nur die Dienstparameter und steuert den Dienstablauf. Teilnehmerdatenverwaltung und Teilnehmerzugang (User Control), sowie die Steuerung der Kommunikationsbeziehungen (Communication Control) und die Verwaltung der Netzressourcen (Resource Control) erfolgen in getrennten Modulen. Aber auch hier ist die Steuerung unabhängig von spezifischen Netzparametern. Adaptoren (Mediation) übernehmen den Zugang zu den unterschiedlichen Kommunikationsnetzen und realisieren die völlige Unabhängigkeit der Steuerplattform von den Netzen und den darin befindlichen Netzressourcen (Networks, Resources). Transparent für die Steuerplattform ist ebenfalls die Ansteuerung von Endgeräten oder Serverelementen.

3.1 Komponenten der Dienstarchitektur

Die Realisierung der oben skizzierten Funktionen in konkreten Komponenten der Dienstarchitektur erfolgt in Anlehnung an Prinzipien der verteilten Systeme. Dabei wird insbesondere auf die TINA Session Konzepte zurückgegriffen [2] (siehe Bild 1, links). In der Teilnehmersteuerung (User Control) vertritt ein User Proxy als Stellvertreter den Teilnehmer in der Zugangs-Session (Access Session). Der Access Session obliegt auch die Verwaltung aller Teilnehmer-spezifischen Daten. Unabhängig von Teilnehmer- und Netz-spezifischen Daten steuert ein Service Session Manager (SSM) einen instanziierten Dienst. Der Communication Session Manager (CSM) wählt je nach Teilnehmerdaten und den Anforderungen des SSM aus der Registry geeignete Adapter aus und steuert darüber die für die Dienstauführung erforderlichen Kommunikationsbeziehungen.

Die eben skizzierte Dienstarchitektur beschreibt eine Serverstruktur aus asynchron kommunizierenden Komponenten, die über Adapter Zugang zu unterschiedlichen Kommunikationsnetzen hat, generell aber unabhängig von einer Netzinfrastruktur ist. Sie beschreibt somit einen unabhängigen Dienstanbieter, der auf verschiedene, ihm in der Regel nicht gehörende, Infrastrukturen für die Erbringung seiner Dienste zugreift. Das Serversystem ist somit nicht in einem Endgerät (Intelligent Terminals) oder in bestimmten netzspezifischen Ressourcen (Intelligent Networks) sondern vielmehr auf einer unabhängigen Dienstebene über den Netzen angeordnet: "Intelligence on Top of the Networks".

Weitere Details zur Steuerung von Diensten und zum Zusammenspiel der Komponenten können [1] entnommen werden. Der Schwerpunkt der vorliegenden Arbeit liegt auf der Realisierung der Netzunabhängigkeit.

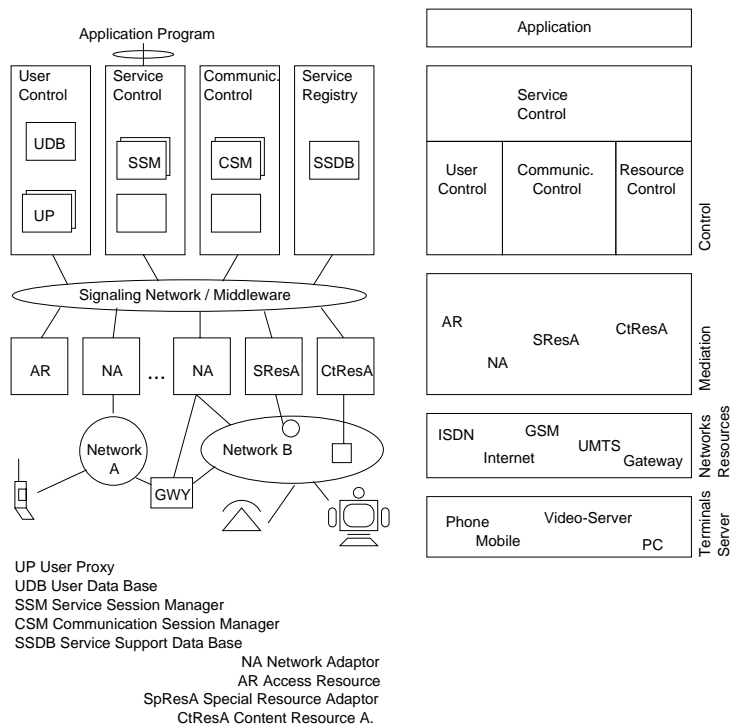


Bild 1: Architektur der netzunabhängigen Dienststeuerung

3.2 Diskussion alternativer Ansätze

Im folgenden Abschnitt werden alternative Ansätze diskutiert, die generell für eine netzunabhängige Dienststeuerung in Frage kommen, und es wird ihr Zusammenhang mit der vorliegenden Arbeit dargestellt.

Das Internet Protokoll (IP) an sich stellt durch seine netzverbindende Eigenschaft eine Plattform für eine Infrastruktur-unabhängige Dienststeuerung dar. Doch zum einen entspricht die endgerätebasierte Realisierung von Diensten nicht unserem Ansatz der Unabhängigkeit von den Endgeräten, zum anderen fehlen notwendige Eigenschaften für Telekommunikationsdienste wie QoS und Vergebührung im IP Transportdienst. Selbst wenn in erweiterten Protokollgenerationen (IPv6) viele Probleme gelöst werden, so gibt es auch dort keine Funktion, die Diensteigenschaften adaptiv auf die Transportnetze abbildet, um deren Eigenschaften auszunutzen.

Middleware-basierte Ansätze wie z.B. die Telecommunications Information Networking Architecture (TINA) [2] sind vielversprechend hinsichtlich der Erbringung komplexer Dienste und erlauben eine Ansteuerung und Auswahl von Netzelementen auf höherer Ebene als beim IP. Während die TINA Kern-Architektur auf eine B-ISDN/ATM Infrastruktur abzielt, werden in weiterführenden Projekten Konzepte für andere

Netze insbesondere bestehende Telekommunikationsnetze [3] und IP basierte Netze entwickelt. Doch stützen sich auch diese Ansätze auf das Vorhandensein der Middleware auf allen beteiligten Netzknoten und Endgeräten. Nicht nur, daß Middleware Technologien wie z.B. CORBA bei weitem noch nicht von allen Endgeräten und Netzknoten unterstützt werden, sondern auch, daß die Performanceprobleme noch nicht endgültig geklärt sind, macht diese Ansätze für unsere Anforderungen ungeeignet. Die abstrakte Modellierung von Diensten in TINA wiederum läßt sich sehr gut übertragen.

Ähnlich wie bei den Middleware-Systemen fordern auch die Ansätze der Programmable Networks, z.B. [4], und der Active Networks [5] standardisierte Interfaces, die in allen beteiligten Netzknoten und teilweise in Endgeräten vorhanden sein müssen. Dies ist bei unserem hier beschriebenen Ansatz nicht notwendig, bei dem die Netzinfrastruktur nicht modifiziert werden muß. Unser Ansatz basiert auf den derzeit entstehenden hochstehenden Steuerungsschnittstellen für Kommunikationsnetze wie z.B. PARLAY [6].

4 Realisierung der Netzunabhängigkeit: Adaptor Units

Die Unabhängigkeit der Dienststeuerung von netzspezifischen Details wird durch Adaptoren erreicht, die als intelligente Signalisierungs-Gateways das Serversystem an Netzelemente koppeln. Wir unterscheiden generell drei Typen von Adaptoren: Network Adaptors (NA), Content Resource Adaptors (CtResA) und Special Resource Adaptors (SpResA). Daneben rechnen wir zwei Funktionseinheiten des Serversystems ebenfalls zur Adaptorebene, die keine echte Gatewayfunktion haben, aber die Adaptivität unterstützen: die Access Resource (AR) für einen Teilnehmerzugang, der unabhängig von der Form der Informationsdarstellung beim Teilnehmer ist, und die Service Registry (SR) zum dynamischen Anbinden von Netzressourcen (= Lookup Service). Im folgenden wird auf die Funktion und die Realisierung der Adaptor-Units näher eingegangen.

4.1 Adaptoren für Kommunikationsnetze und -ressourcen

Die Network Adaptors (NA) stellen einen Zugang zu Kommunikationsnetzen bereit und ermöglichen die Einrichtung von Kommunikationsbeziehungen in denselben. Dabei können die NAs auf hochstehende, mächtige 3rd Party Interfaces zugreifen wie z.B. PARLAY [6], oder die architektureinheitliche Signalisierung wird auf einfache UNI Schnittstellen umgesetzt. Für Kommunikationsbeziehungen die nicht stellvertretend (3rd Party) aufgebaut oder über sequentiell gekoppelte UNIs gesteuert werden können, sind NAs vorgesehen, die über Gateways Kommunikationsbeziehungen vom Netzrand her einrichten (Network Edge Initiated, z.B. Internet Gateways, DVB-T Kopfstation). Um die Funktionalität mächtiger Schnittstellen auszunutzen, werden Kommunikationsbeziehungen möglichst abstrakt auf ein Netz abgebildet. Beispiel: Falls ein Interface bereits die Einrichtung von Videokonferenzen erlaubt, wird dies ausgenutzt und es entfällt die komplexe Aufteilung in und Koordinierung von einzelnen Point-to-Point Verbindungen.

Neben den NA bieten die SpResA Schnittstellen zu Ressourcen in den Netzen wie

z.B. Konvertern, Mischern, etc. CtResA bieten ergänzend Schnittstellen zu Servern in den Netzen, die Inhalte bereitstellen, z.B. Ansageeinheiten, Videosever, Game Server. Die Signalisierungsplattform bleibt damit unabhängig von Informationsinhalten (Content).

Jeder Adaptor bildet die architektureinheitliche Signalisierung auf die jeweilige, netzspezifische Signalisierung ab und kontrolliert eine Kommunikationsbeziehung in seinem Netz. Dafür besteht er aus einer generischen Einheit zum Protokollhandling (Protocol Agent), einer spezifischen Komponente und einer Abbildungseinheit dazwischen. Mögliche standardisierte oder proprietäre Netz-APIs oder Netz-Komponenten, auf denen die Adaptoren aufsetzen, sind: UNI (ISDN), INAP (IN), PINT (IN-SCP), PARLAY, JTAPI, JAIN (ISDN), Gatekeeper (H.323), Proxy Server (SIP). Daneben können auch beliebige neue Adaptoren entwickelt werden, um neue Netze anzubinden, insbesondere auch Broadcast-Netze.

Die Verwendung der Datenübertragungseigenschaften von terrestrischen Broadcast-Netzen wie z.B. DVB-T und DAB verspricht eine verbesserte Unterstützung für breitbandige, mobile Dienste. Zur Datenübertragung über DVB-T wurde im Rahmen der Dienstarchitektur ein System entwickelt und realisiert, das sowohl die unidirektionale Datenübertragung im Broadcast oder Multicastbetrieb als auch eine Kombination mit schmalbandigen Rückkanälen erlaubt [7]. Die Ansteuerung dieses Broadcast Netz Adaptors verspricht für die vorgestellte Dienststeuerung neue Möglichkeiten zur Dienstrealisierung.

4.2 Service Mobility: Flexibler Dienstzugang mit XML

Die Signalisierung zwischen den Teilnehmern und dem Serversystem ist bei den oben vorgestellten Adaptoren beschränkt auf die Möglichkeiten, die die netzspezifische Signalisierung bietet. Gerade neue Dienste können aber damit nicht oder nur sehr umständlich angeboten werden. Darüberhinaus erlaubt ein signalisierungstransparenter Zugang einen Netz- und Endgeräte-unabhängigen Dienstzugriff (Service Mobility). Unser Konzept sieht daher die Verwendung eines zusätzlichen Adaptors vor, der die Darstellung der dienstspezifischen Signalisierung für den Teilnehmer übernimmt: die Access Resource (AR).

Wir verwenden dabei eine flexible Hypertext-basierte Lösung. Der Inhalt des Hypertextservers, der je nach Signalisierung des Serversystems dem Teilnehmer angezeigt wird, ist in XML gespeichert bzw. es werden über Servlets dynamisch XML Seiten aus einer LDAP Teilnehmerdatenbank erzeugt. Durch die Verwendung von XML wird eine darstellungsunabhängige Beschreibung des anzuzeigenden Inhalts erreicht. Diese kann adaptiv je nach Fähigkeiten des Teilnehmerendgerätes z.B. in HTML für Multimedia Endgeräte, in WML für WAP Handys oder in VoxML für die Sprachausgabe auf Telefonen ausgegeben werden. Das Darstellungsformat wird in Style Files (XSL) festgelegt. Diese bestimmen auch, welche Inhalte in der jeweiligen Darstellung sinnvoll angezeigt werden. Abbildung 2 gibt einen Überblick über die prototypische Implementierung der AR. Für die Realisierung wird unter Linux ein Apache Web-Server mit dem COCOON Plug-In für die Verarbeitung von XML eingesetzt.

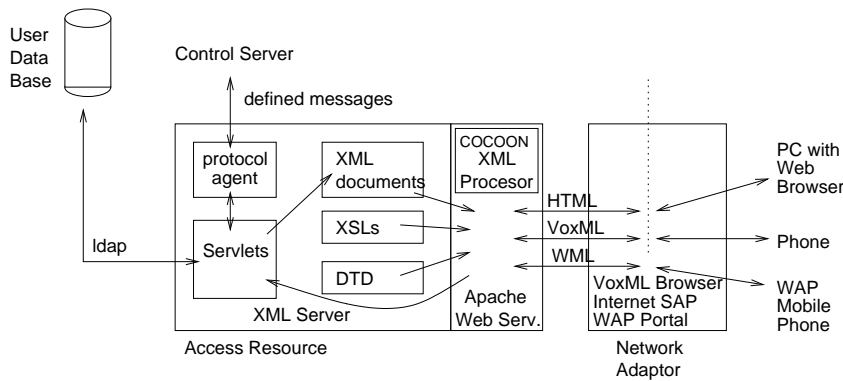


Bild 2: Darstellungsunabhängiger Dienstzugang

4.3 Service Discovery: Adaptive Auswahl von Netzressourcen

Um zur Laufzeit auf Änderungen in der Dienstarchitekturumgebung reagieren zu können, beinhaltet das Serversystem eine sogenannte Service Registry als weitere Adaptorkomponente. Diese unterstützt die Kommunikationssteuerung (Communication Control) in der Auswahl geeigneter Netz- und Ressourcenadaptoren. Dabei erlaubt ein Mechanismus aus den bekannten Service Discovery Ansätzen die Registrierung der Kommunikationsdienste der Adaptoren in der Registry (siehe Bild 3). In der Registry können neben der (internen) Adresse und der angebotenen Netz-Dienstparametern auch weitere Parameter, wie z.B. Preise abgespeichert werden. Darüberhinaus ist es angedacht auch dynamische Parameter wie z.B. die momentane Auslastung oder die Verfügbarkeit (bei Mobilnetzen) zu speichern. Die Realisierung der Service Discovery Komponenten in unserem Prototypen basiert auf SUN's JINI Technologie.

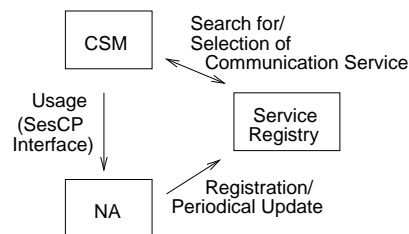


Bild 3: Service Discovery für die Ressourcenverwaltung

5 Server Middleware: SIP basierte Signalisierung

Durch die strenge Separierung in getrennte Funktionseinheiten für Teilnehmerdaten, Dienststeuerung und Kommunikationssteuerung wird die Komplexität für die Steue-

rung von Diensten reduziert. Darüberhinaus können die verschiedenen Komponenten der Dienststeuerung verteilt angeordnet werden. Dies wird insbesondere durch die gewählten Schnittstellen und den darauf definierten Meldungssatz unterstützt über den die verteilten Komponenten asynchron Nachrichten austauschen. Der Meldungssatz ist dabei möglichst einfach gehalten, um zum einen eine unkomplizierte Realisierung zu ermöglichen und um zum anderen Flexibilität zu schaffen, d.h. unabhängig zu sein von der Implementierung bestimmter Dienstklassen.

Das Session Initiation Protocol (SIP) der IETF [8] bietet die hierfür geforderte Einfachheit in Kombination mit Flexibilität d.h. Offenheit zur Unterstützung verschiedener Dienste. Wir haben auf Basis von SIP das *Session Control Protocol* (SesCP) für die Signalisierungsplattform entwickelt. Tabelle 1 gibt einen Überblick über die verwendeten Meldungen und stellt den Bezug zu SIP dar.

Tabelle 1

Meldung	Kurzbeschreibung	Bezug zu SIP
ACCESS	Teilnehmerzugang zur Dienststeuerung und Management des Teilnehmerprofils	REGISTER
INVITE	Anforderung eines Dienstes und Änderung der Dienstkonfiguration (Einladen eines Tln., etc.)	INVITE
SETUP	Anforderung der Einrichtung von Kommunikationsbeziehungen	INVITE
OK	Antwort auf INVITE / SETUP	OK (1xx)
ACK	Bestätigung von OK und Ausführungsaufforderung	ACK
INFO, STATUS	Abfrage / Mitteilung von Informationen (z.B. Teilnehmerprofil)	-
END, BYE	Beendigung der Teilnahme (BYE) oder des ganzen Dienstes (END)	BYE
CANCEL	Vorzeitiger Abbruch einer Dienstanforderung	CANCEL

Im Session Control Protocol wird auch die aus TINA abgeleitete Separierung in die drei Steuerungsbereiche User, Service und Communication Control (siehe Abschnitt 3) unterstützt. Jeder Steuerungsbereich verwaltet, wie bereits beschrieben, seine eigene Sicht auf den Dienst (Dienstbeschreibung). Die SesCP Meldungen kommunizieren diese Bereichs-spezifischen Dienstbeschreibungen oder Teile davon zwischen den Architekturkomponenten, um die verschiedenen Sichten auszutauschen. Dabei wird die Dienstbeschreibung beginnend von einem abstrakten Modell (Teilnehmersicht: Dienstname und Teilnehmer) schrittweise detailliert, bis schließlich konkrete Kommunikationsbeziehungen in den Netzadaptoren eingerichtet werden (QoS Parameter, Verbindungsgraph). Die Hauptkomponenten der drei Steuerungsbereiche (UP, SSM, CSM) wirken dabei als (SIP-) Proxy Server zur Verarbeitung und Weiterleitung der Dienstbeschreibung.

Abbildung 4 zeigt einen typischen Meldungs austausch zur Einrichtung von Kom-

munikationsbeziehungen zwischen der Dienststeuereinheit (SSM) und der Kommunikationssteuerung (CSM). Dabei wird zum einen der Einsatz des dreischrittigen Meldungsablaufes (INVITE, OK, ACK) nach dem Atomic Action Prinzip deutlich [9]. Das dreischrittige Protokoll erlaubt das Zusammenfügen verschiedener Steuereinheiten (NA, SpResA) zu einem Verbindungsgraphen und dessen Aushandlung. Durchgeschaltet wird erst, wenn alle Einheiten positiv geantwortet haben (OK).

Zum anderen kommt die strikte Trennung von Dienst und Teilnehmerdaten zum Ausdruck. Erst der CSM fordert Teilnehmerdaten wie aktueller Netzzugang an (INFO). Die initiale SETUP Meldung des SSM enthält nur abstrakte Parameter, auf

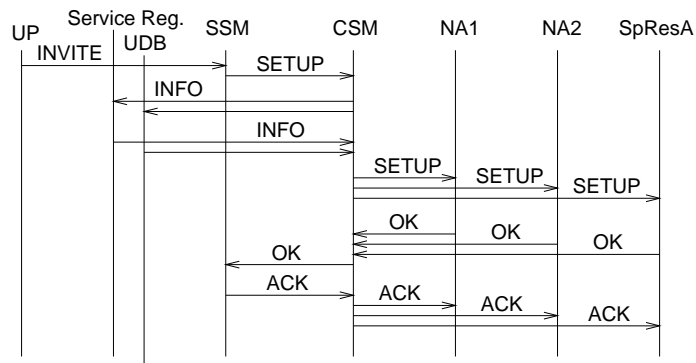


Bild 4: Anforderung von Kommunikationsbeziehungen

deren Basis der CSM die Netze bzw. deren Adaptern auswählt.

Eine prototypische Realisierung der Signalisierungsarchitektur erfolgt derzeit mit der formalen Beschreibungssprache SDL.

6 Zusammenfassung und Ausblick

In der vorliegenden Arbeit wird eine neuartige Architektur zur netzübergreifenden Steuerung von Informations- und Kommunikationsdiensten unabhängig von Netzen und Endgeräten vorgestellt. Auf diese Weise wird der zunehmenden Heterogenität einerseits und dem Verlangen nach personalisierten, netzunabhängigen Diensten andererseits Rechnung getragen. Der Schwerpunkt der Ausführungen liegt auf den Komponenten, die im Sinne einer auf asynchronem Nachrichtenaustausch basierenden Middleware die Netzunabhängigkeit realisieren. Neben Adaptor-Komponenten für die Anpassung von Signalisierungsnachrichten werden Komponenten für einen darstellungsunabhängigen Dienstzugang beschrieben. Adaptivität bezüglich sich ändernder Netzinfrastruktur wird durch Service Discovery Mechanismen unterstützt.

Die vorgestellte Architektur ermöglicht es in Weiterführung des Abstraktionsgedankens, der mit der Einführung von mächtigen APIs wie z.B. dem PARLAY API aufgekommen ist, Dienste netzübergreifend zu steuern. Damit haben auch reine Dienstanbieter die Möglichkeit, neue Dienste schnell, Netz- und Endgeräte-übergreifend anzubieten. Durch die Verwendung eines Signalisierungsprotokolls, das auf dem

IETF Protokoll SIP basiert, wird der Übergang auf ein IP basiertes Next Generation Network vereinfacht, indem die Einbeziehung von vorhandener Infrastruktur in zukünftige IP basierte Netze über die vorgestellte Signalisierungsplattform erreicht werden kann.

7 Literatur

- [1] W. Kellerer. *A Versatile Network Independent Server Architecture for Multimedia Information and Communication Services*. In: H. van As (Editor) *Telecommunication Network Intelligence*, Kluwer, 2000. Proceedings of SmartNet 2000, Vienna, September 2000, pp. 331-350.
- [2] *TINA Service Architecture*, Telecommunications Information Networking Architecture Consortium, [Http://www.tinac.com](http://www.tinac.com), 1997.
- [3] E. Holz, O. Kath, M. Geipl, G. Lin, and V. Vogel. *The CAMOUFLAGE Project: Introduction of TINA into Telecommunication Legacy Systems*. In Proceedings of TINA'97, November 1997, pp. 206-215.
- [4] C. Gbgaguidi, J.-P. Hubaux, M. Hamdi, and A. Tantawi. *A Programmable Architecture for the Provision of Hybrid Services*. *IEEE Communications Magazine*, Vol. 37, No.7, July 1999, pp. 110-116.
- [5] K. L. Calvert, S. Bhattacharjee, E. Zegura, G. Tech, J. Sterbenz. *Directions in Active Networks*. *IEEE Communications Magazine*, October 1998, pp. 72-78.
- [6] F.-K. Bruhns. *Parlay - the API for Secure and Open Access to Networking Functionality for Third Party Applications outside the Network*. In Proceedings of ICIN'2000, Bordeaux, January, 2000, pp. 17 - 21.
- [7] W. Kellerer, P. Sties and J. Eberspächer. *IP Based Enhanced Data Casting Services over Radio Broadcast Networks*. In Proceedings of IEEE ECUMN'00, Colmar, France, October 2000, pp. 195-203.
- [8] M. Handley, H. Schulzrinne, E. Schooler, and J. Rosenberg. *SIP: Session Initiation Protocol*. IETF Request for Comments, RFC 2543, March 1999.
- [9] ITU-T Recommendations X.851/2. *Information Technology Open Systems Interconnection - Protocol For The Commitment, Concurrency And Recovery Service Element*. ITU-T, 1993.