



TECHNISCHE UNIVERSITÄT ILMENAU

Fakultät für Elektrotechnik und Informationstechnik

Diplomarbeit

Next Generation Network (NGN)

Aufbau eines Anwendungsservers für NGN

Inventarisierungsnummer: 2009-12-08 / 119 / II03 / 2115

vorgelegt von:	Torsten Strelle
eingereicht am:	07.12.2009
geboren am:	07.12.1983 in Borna
Studiengang:	Ingenieurinformatik
Studienrichtung:	Informatikkomponenten für Intelligente Systeme
Anfertigung im Fachgebiet:	Kommunikationsnetze Fakultät für Elektrotechnik und Informationstechnik
Verantwortlicher Professor:	Prof. Dr. rer. nat. habil. Jochen Seitz
Wissenschaftlicher Betreuer:	Dipl.-Ing. Yevgeniy Yeryomin

Danksagung

Ich möchte mich an dieser Stelle bei all denen bedanken, die mich bei der Erstellung dieser Arbeit unterstützt haben.

Mein herzlicher Dank gilt Herrn Prof. Dr. Jochen Seitz für die tatkräftige und kritische Betreuung der Diplomarbeit. Besonderen Dank schulde ich Herrn Yevgeniy Yeryomin, der mich durch seine engagierte Betreuung und stete Diskussionsbereitschaft, auch auf große Entfernung, immer mit vielseitigen Denkanstößen bereicherte und bei der Erstellung dieser Arbeit unterstützt hat.

Vielen Dank auch an Kristin Krüger, die mir als Lektorin viele wertvolle Hinweise zur Korrektur der Arbeit geliefert und mit ihrer unendlichen Geduld letztendlich sehr zur Fertigstellung beigetragen hat. Nicht zuletzt möchte ich meinen Eltern danken, die mir durch ihre fortwährende Unterstützung das Studium und diese Arbeit ermöglichten und sie mit Anteilnahme verfolgt haben.

Kurzfassung

Das Next Generation Network (NGN) ist ein IP-basierendes Telekommunikationsnetz, das entwickelt wurde um einen nahtlosen Übergang von den herkömmlichen Telefonnetzen zu ermöglichen. Das NGN vereint eine Vielzahl von Zugangsnetzen, die für den Nutzer über ein einheitliches Netzmodell wie ein einzelnes Netz nutzbar sind. Mit der Einführung des NGN werden auch neue Leistungsmerkmale in den Telekommunikationsnetzen verfügbar. Sprachkommunikation ist dabei nur noch eine Dienstleistung unter vielen, da andere Medien, wie Videokommunikation, Dateitransfer, Textnachrichten oder Chat, genauso für alle Netzteilnehmer nutzbar sind.

Durch diese Veränderungen werden neue Mehrwertdienste im NGN möglich, welche auf der Grundlage der verfügbaren Medien neue Funktionen bereitstellen. Nach der Einführung in das Modell des NGN werden dieser Arbeit Konzepte für solche innovative Mehrwertdienste untersucht und bewertet. Anschließend erfolgt eine Betrachtung der standardisierte Schnittstellen und Plattformen zur Bereitstellung solcher Dienste. Abschließend wird eine dieser Plattformen in ein NGN integriert und zur prototypischen Umsetzung einiger Dienstkonzepte eingesetzt.

Abstract

The Next Generation Network (NGN) is an IP-based telecommunications network which was aimed to provide a seamless transition from traditional telephone networks. The NGN combines a variety of access networks which are available to users as a single network by a integrative network model. Together with the introduction of NGN, new features came into telecommunication networks. Voice is only one among many services, beside other media like video communication, file transfer, text messaging or chat room, which are now available to all agents.

These changes enable new value-added services in the NGN, which provide their services based on the new features. After an the introduction into the model of the NGN, this work will examine such innovative approaches to value-added services. This is followed by a discussion of standardized interfaces and platforms for the implementation of such services. Finally, one of these platforms will be integrated into the NGN and used for the prototype implementation of some service concepts.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Ziel der Arbeit	2
2	Stand der Technik	3
2.1	Die heutigen Telekommunikationsnetze	3
2.1.1	Festnetz (ISDN)	4
2.1.2	Mobilfunknetz 2G (GSM)	6
2.1.3	Mobilfunknetz 3G (UMTS)	8
2.2	Intelligentes Netz - Dienste im ISDN	9
2.2.1	Klassifikation der Dienste	10
2.3	Digitale Breitbandnetze	12
2.4	Motivation für Netz der nächste Generation	14
3	Spezifikationen für Next Generation Networks	15
3.1	Übersicht über Standardisierungsbemühungen	16
3.1.1	ITU - Mobilität in NGNs	16
3.1.2	3GPP - NGNs auf Basis von Mobilfunknetzen	19
3.1.3	ETSI TISPAN - NGNs als Ersatz für ISDN	21
3.1.4	Zusammenfassung	23
3.2	Technische Grundlagen	24
3.2.1	Netzwerkprotokolle	24
3.2.1.1	Internet Protocol (IP)	24
3.2.1.2	Transmission Control Protocol (TCP)	24
3.2.1.3	User Datagram Protocol (UDP)	25
3.2.1.4	Sitzungsprotokolle	25
3.2.2	SIP - Session Initiation Protokoll	26
3.2.2.1	SIP URI	27
3.2.2.2	SIP Nachrichten	28

3.2.2.3	Verbindungsaufbau	30
3.2.3	SDP - Session Description Protocol	32
3.2.4	Codecs	33
3.2.5	Real-Time Transport Protocol (RTP)	34
3.3	Netzelemente	35
3.3.1	User Agent	35
3.3.2	Registrar Server	36
3.3.3	Proxy Server	36
3.3.4	Location Server (HSS)	37
3.3.5	Gateways	38
3.3.6	Back-to-Back User Agent	38
3.3.7	Konferenzserver	39
3.3.8	Presence Server	40
3.3.9	Anwendungsserver	41
3.3.10	weitere Netzelemente	42
3.4	Zugangsnetze	42
3.5	Netzmerkmale	43
3.5.1	Dienstgütegarantien (QoS)	43
3.5.1.1	ATM	44
3.5.1.2	MPLS	44
3.5.1.3	IntServ und DiffServ	44
3.5.2	Mobilität	45
3.5.2.1	Persönliche Mobilität	45
3.5.2.2	Endgerätemobilität	45
3.5.2.3	Sitzungsmobilität	46
3.5.2.4	Dienstemobilität	46
3.5.3	Sicherheit	46
3.5.4	Dienste	47
3.6	Zusammenfassung	47
4	Dienste im NGN	49
4.1	Definition (Dienst)	49
4.2	Klassifizierungsschemata für Dienste	50
4.2.1	Klassifizierung nach Art der Dienstleistung	52
4.2.2	Klassifizierung nach Funktion des Dienstes	53
4.2.3	Klassifizierung nach Betriebsmodus des Anwendungsservers	54

4.2.4	Klassifizierung nach genutzten Leistungsmerkmalen des Netzes	55
4.3	Mehrwertdienste im NGN	57
4.3.1	Erreichbarkeitsdienste	57
4.3.2	Videostreamingdienste	63
4.3.3	Multimedienadienste	66
4.3.4	Alarmierungsdienste	74
4.3.5	PIM-Dienste	76
4.3.6	Informationsdienste	77
4.3.7	(Haushalts-) Steuerungsdienste	78
4.3.8	Location-based Services (LBS)	80
4.4	Bewertung und Ausblick	83
5	Anwendungsserver für NGNs	85
5.1	Anwendungsserver als Teil des Netzes	85
5.1.1	SDP - Service Delivery Platform	85
5.1.2	Definition Anwendungsserver	86
5.1.3	Anwendungsserver als Netzelement	86
5.1.4	Betriebsmodi eines Anwendungsservers	87
5.2	Technische Konzepte	91
5.2.1	High-Level vs. Low-Level-APIs	91
5.2.2	High-Level-Lösungen	92
5.2.2.1	Parlay/OSA	92
5.2.2.2	OMA Service Environment (OSE)	93
5.2.2.3	Camel Service Environment (CSE)	94
5.2.2.4	Java APIs for Integrated Networks (JAIN)	94
5.2.3	Low-Level-Lösungen	95
5.2.3.1	SIP-Servlets	95
5.2.3.2	Call Processing Language (CPL)	95
5.2.3.3	SIP-CGI	96
5.2.4	Zusammenfassung und Bewertung	96
5.3	Verfügbare Softwarelösungen	97
5.3.1	Frauenhofer Open Source IMS Playground	97
5.3.2	IBM WebSphere IMS Connector	98
5.3.3	Mobicents	99
5.3.4	OpenCloud Rhino Application Server	100
5.3.5	Ericsson IMS/ SailFin	101

5.4	Client-Seitige Applikationen	101
5.5	Bewertung und Auswahl	102
6	Konzept für Netzaufbau	105
6.1	Ausgangssituation	105
6.2	Netzkonzept	106
6.2.1	Anforderungen	106
6.2.2	Verwendete Netzelemente	107
6.2.2.1	User Agents	107
6.2.2.2	Sitzungs- und Verbindungssteuerung	109
6.2.2.3	Gateways	109
6.2.2.4	Registrar Server und HSS	109
6.2.2.5	Anwendungsserver	109
6.3	Realisierung	110
6.3.1	Asterisk PBX als zentrales Element	110
6.3.1.1	Aufbau	110
6.3.1.2	Kommunikation	110
6.3.1.3	Funktionalitäten	111
6.3.1.4	Dienste	112
6.3.1.5	Beschränkungen	112
6.3.1.6	Alternative PBX-Software	112
6.3.2	SipToSis - Anbindung an Skype	113
6.3.3	Anwendungsserver vs. Asterisk API	114
6.4	Prototypisches Netz	115
7	Praktische Umsetzung	117
7.1	Asterisk	117
7.1.1	Dialplan	117
7.1.2	Asterisk Gateway Interface (AGI)	120
7.2	Mobicents Anwendungsserver	121
7.2.1	Aufbau	121
7.2.2	Funktionsweise	122
7.2.3	Verwaltung	123
7.3	Dienstkonzepte	125
7.3.1	Ring Back when Free (RBWF)	125
7.3.2	Combinatory Services	126
7.3.3	Click-to-Dial	127

7.3.4	Mediencontroller	129
7.3.5	IPTV Call Notifier	130
7.3.6	Veranstaltungskalender	132
7.3.7	Netzbasierende Kontaktliste	133
7.4	Ausgangspunkt für weitere Dienste	135
7.4.1	Mitgelieferte Resource-Adapter	135
7.4.2	Kombination von Funktionsblöcken	136
8	Fazit	139
8.1	Zusammenfassung	139
8.2	Ausblick	140
A	Software und Konfigurationsdateien	143
A.1	Einrichtung Asterisk	143
A.1.1	Anbindung an Sipgate	143
A.1.2	Anbindung an Skype (SipToSis)	145
A.2	Einrichtung Mobicents	146
A.2.1	Installation des Java JDK/JRE	147
A.2.2	JAVA-Umgebungsvariable	147
A.2.3	Auswahl der JVM	147
A.2.4	Paketauswahl und Download	148
A.2.5	Konfiguration	148
A.2.6	Starten	148
A.3	Setup der Entwicklungsumgebung	149
A.3.1	Installation benötigter Tools	149
A.3.1.1	Apache Ant	149
A.3.1.2	Apache Maven	149
A.3.1.3	JBoss Anwendungsserver	150
A.3.1.4	Entwicklungsumgebung Eclipse	151
A.3.1.5	SVN-Client	151
A.3.2	Installation Mobicents aus Quelltext	152
A.3.3	Einrichtung Eclipse	153
A.3.4	Eclipse-Plugins	155
A.3.5	Anlegen von Beispielen	155
B	Abbildungen	157

Literaturverzeichnis	159
Standards und Empfehlungen	167
Abbildungsverzeichnis	171
Tabellenverzeichnis	175
Abkürzungsverzeichnis und Formelzeichen	177
Thesen zur Diplomarbeit	181
Erklärung	183

1 Einleitung

1.1 Motivation

Die herkömmlichen Telefonnetze sind für die statischen Anforderungen der Sprachkommunikation ausgelegt, woran sich in den letzten 100 Jahren kaum etwas geändert hat [Sieg02, Vorwort]. Durch die hohe Verfügbarkeit und die weltumspannende Eigenschaft der Kommunikationsnetze sind Veränderungen sehr aufwendig und kostspielig, wodurch die Telefonnetze wiederum einer deutlich niedrigeren Entwicklungsgeschwindigkeit unterworfen sind als der Rest der Technik.

Die Datenkommunikation über das Telefonnetz und in separaten Netzen hat seit der Digitalisierung der Netze ständig zugenommen und stellt mittlerweile einen Großteil des gesamten Verkehrsaufkommens dar. Damit ist die Konvergenz von Daten- und Telefonnetz zu einem gemeinsamen *Next Generation Network* (NGN) eindeutig der nächste Schritt in der Evolution der Kommunikationsnetze. Durch die damit verbundenen strukturellen Veränderungen stehen für die Realisierung von Diensten im Next Generation Network sehr viele neue Möglichkeiten zur Verfügung.

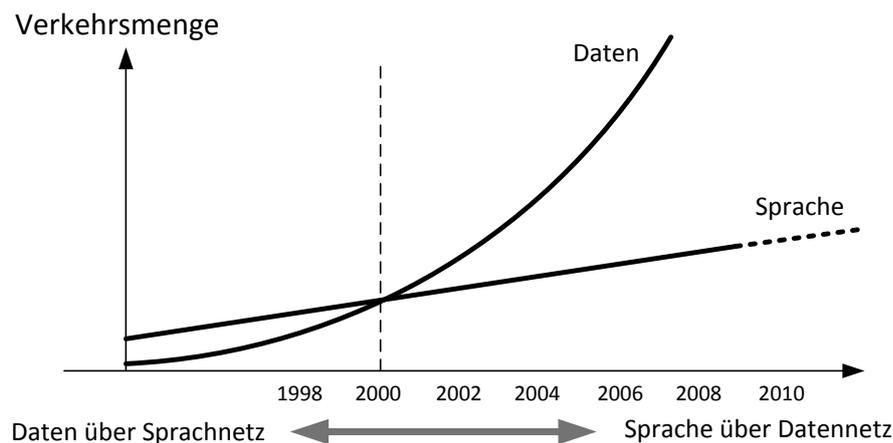


Abbildung 1.1: Quantitative Entwicklung der Sprach- und Datenkommunikation, von der Datenübertragung über Sprachnetze hin zur Sprachübertragung in reinen Datennetzen. (Quelle: [Sieg02, S.2])

1.2 Ziel der Arbeit

Die Konvergenz von Daten- und Telekommunikationsnetzen bedeutet, dass die heute etablierten heterogenen, drahtgebundenen und mobilen, leitungs- und paketvermittelnden Netze zusammengefasst werden. Sie bilden aus Sicht des Nutzers ein einheitliches Netz, das er mit verschiedenen Endgeräten oder einem universellen Endgerät nutzen kann.

Aus heutiger Sicht erscheint es sinnvoll, alle Dienste über ein IP-Netz abzuwickeln, das mit geeigneten Mechanismen zur Sicherung der jeweils benötigten Dienstgüte ausgestattet ist. Ein solches „Next Generation Network“ wird von allen Netzbetreibern mit entsprechenden Migrationswegen anvisiert. [Tric03, S.30]

Der Fokus der Arbeit liegt auf der Untersuchung von neuen Möglichkeiten für Mehrwertdienste im NGN. Dazu werden, nach einem Überblick über die aktuellen Telekommunikationsnetze, in Kapitel 3 die Standardisierungsbemühungen, das resultierende Modell und die technischen Grundlagen für das NGN untersucht. Mit der Nutzung der bestehenden paketbasierenden Datennetze als Basis für dieses neue Netz sind alle technischen Möglichkeiten, die heute beispielsweise im Internet zum Einsatz kommen, auch im NGN verfügbar.

Konzepte und konkrete Beispiele für solche innovativen Mehrwertdienste sind in Kapitel 4 dargestellt, wobei auch auf eine mögliche Realisierung dieser Dienste näher eingegangen wird. Für die Umsetzung dieser Dienste existiert bereits eine Reihe von standardisierten Schnittstellen und Plattformen, die im Zuge der NGN-Spezifikation entworfen wurden und in Kapitel 5 genauer betrachtet werden.

Zum Abschluss werden in Kapitel 6 der Aufbau eines lokalen Next Generation Network besprochen und die prototypische Umsetzung einiger Mehrwertdienste auf Basis eines Anwendungsservers in diesem Netz in Kapitel 7 vorgestellt.

2 Stand der Technik

2.1 Die heutigen Telekommunikationsnetze

Ein Telekommunikationsnetz bildet die materielle Grundlage einer Telekommunikationsinfrastruktur. Es besteht im Allgemeinen aus Übertragungssystemen mit Übertragungswegen, Vermittlungssystemen und Endeinrichtungen [TrWe07, S. 4].

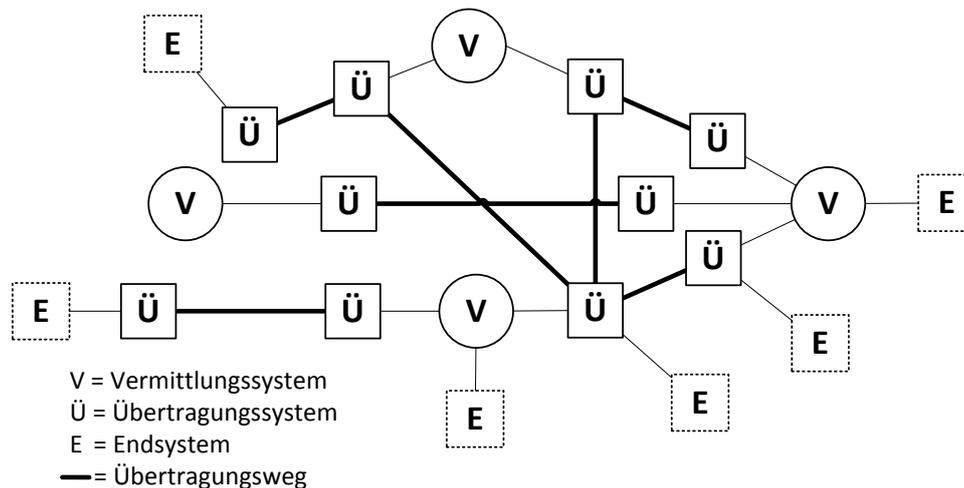


Abbildung 2.1: Allgemeine Struktur eines Telekommunikationsnetzes (nach [TrWe07, S.5])

In Abbildung 2.1 ist ein exemplarisches Telekommunikationsnetz dargestellt. Die Übertragungssysteme bilden zusammen mit ihren Verbindungen, den Übertragungswegen, das Kernnetz. Die Endeinrichtungen ermöglichen den Nutzern den Netzzugang, und sind über Vermittlungssysteme an das Kernnetz angebunden. Diese Vermittlungssysteme sind für das Routing der Informationen im Netz zuständig. Der Weg von den Endeinrichtungen bis zu den Vermittlungssystemen wird als Zugangnetz bezeichnet.

Unterschiede zwischen verschiedenen Netzen können dabei besonders in der Art der Nachrichtenvermittlung bestehen [Sieg01, S.7]. Bei einer verbindungsorientierten Nachrichtenübertragung wird ein exklusiver Übertragungskanal zwischen zwei Endsystemen bereitgestellt. Dieses Verfahren wird in allen leitungsvermittelnden Netzen verwendet

und garantiert für die Dauer der Verbindung eine festgelegte Übertragungsqualität. Bei der verbindungslosen Nachrichtenübertragung werden die Daten als Pakete durch das Netz gesendet, ohne dass vorher ein Verbindungsaufbau stattfindet. Diese Pakete werden von den Übertragungssystemen spontan weitergeleitet, wobei sich der Übertragungsweg aufeinanderfolgender Pakete unterscheiden kann und keine Garantien über die Übertragungseigenschaften abgegeben werden können (*Best Effort*).

2.1.1 Festnetz (ISDN)

Das herkömmliche Telefonnetz (PSTN¹) stellt ein reines leitungsvermittelltes Telekommunikationsnetz dar. Die Infrastruktur ist ursprünglich für analoge Sprachverbindungen in einem Frequenzbereich von $0,3 - 3,4\text{kHz}$ ausgelegt. Diese Nutzung dieser analogen Verbindungen wird heute als *Plain Old Telephone Service* (POTS) bezeichnet.

Eine Verbindung bestand aus einer exklusiven direkten elektrischen Verbindung zwischen zwei Teilnehmern. Diese wurde über einen Vermittlungsdienst manuell und später über sogenannte Selbstwähleinrichtungen automatisch hergestellt. Als Endgeräte waren nur analoge Telefone möglich. Da das Netz in sehr engen Grenzen auf die Sprachübertragung ausgelegt war, wurde für andere Dienste wie Fernschreiber ein separates Netz mit eigenen Vermittlungssystemen genutzt.

Um 1970 wurde erstmalig die Digitalisierung der Vermittlungstellen im Telefonnetz in Erwägung gezogen, um die mechanischen Vermittlungsstellen zu ersetzen. Die Digitalisierung des Telefonnetzes bot gleichzeitig die Chance mit einer Art universellen Kommunikationssteckdose [KaKö91] verschiedene Systeme und Dienste über ein einheitliches Netz zu verbinden. Die Dienstleistungen, die dafür vorgesehen waren, sind beispielsweise:

- Telefon
- Bildtelefon
- Telefax (auch Fernkopierer)
- Fernschreiben (Teletex)
- Informationsabruf
- Datenübertragung
- Fernwirken und -messen.

¹Public Switched Telephone Network (PSTN) bezeichnet ganz allgemein das leitungsgebundene Telefonnetz.

Aus diesem Grund wird das digitale Telefonnetz international als *Integrated Services Digital Network* (ISDN) bezeichnet, da es zum Zeitpunkt der Einführung das erste Telekommunikationsnetz war, was mehrere Dienste (Sprache und Daten) über die gleiche Verbindung ermöglichte. Die im Zuge der Digitalisierung eingeführten neuen Dienste wurden unter dem Stichwort *Intelligentes Netz* (IN) vermarktet (siehe Kapitel 2.2).

Bei Feldversuchen der Deutschen Bundespost in Berlin ab 1979 konnte nachgewiesen werden, dass durch die digitale Übertragung über eine herkömmliche Doppelader der analogen Teilnehmeranschlussleitung zwei unabhängige Duplexkanäle gleichzeitig übertragen werden können. Im Jahr 1982 wurde endgültig entschieden, dass ein digitales Telefonnetz eingeführt werden soll, woraufhin in Kooperation mit der Industrie international ein Konzept und Spezifikationen für ein digitales Telefonnetz entworfen wurde. Parallel wurden 1984 die ersten internationalen Empfehlungen veröffentlicht, die ebenfalls eingeflossen sind. Das erste Pilotprojekt wurde 1987 gestartet.

Da rechtzeitig erkannt wurde, dass in Europa eine einheitliche Telekommunikationsinfrastruktur nötig war, um international nicht den Anschluss zu verlieren, wurde 1988 das *European Telecommunications Standards Institute* (ETSI) gegründet. Dieses Institut verabschiedete am 6. April 1989 den Signalisierungsstandard DSS1², welcher für die nationalen digitalen Telefonnetze eine einheitliche Grundlage bilden sollte. Daraufhin wurde 1989 die erste Stufe des ISDN-Netzes in Deutschland als erstem europäischen Land offiziell in Betrieb genommen.

Strukturell gleicht das ISDN-Netz damit dem analogen Telefonnetz, da besonders die Teilnehmeranschlüsse („letzte Meile“) eine erhebliche Investition darstellen und dementsprechend weiter genutzt werden sollten. Das Netz unterteilt sich in viele Ortsnetze, welche die Teilnehmeranschlüsse einer geografischen Region zusammenfassen, und ein Fernnetz, welches die einzelnen Ortsnetze miteinander verbindet (siehe Abb. 2.2).

Mit der Einführung von ISDN können die einzelnen Teilvermittlungsstellen (TVst) mehrere Anschlussbereiche mit unterschiedlichen Nummerierungen bedienen. Eine TVst kann etwa 100.000 Teilnehmer verwalten, welche bei größeren Entfernungen auch über Konzentratoren oder abgesetzte Anlagenteile mit der TVst verbunden sein können [Sieg02, S.42ff]. Die Teilnehmer können dabei über eine Teilnehmerrufnummer, der eine Bereichskennzahl (National Destination Code) und eine Länderkennzahl (Country Code) vorangestellt sind, erreicht werden.

Das Fernvermittlungsnetz leitet Anrufe nur noch anhand der Länderkennzahlen und

²Digital Subscriber Signalling System No. 1 (DSS1) - beschreibt das Signalisierungsprotokoll des ISDN B-Kanals.

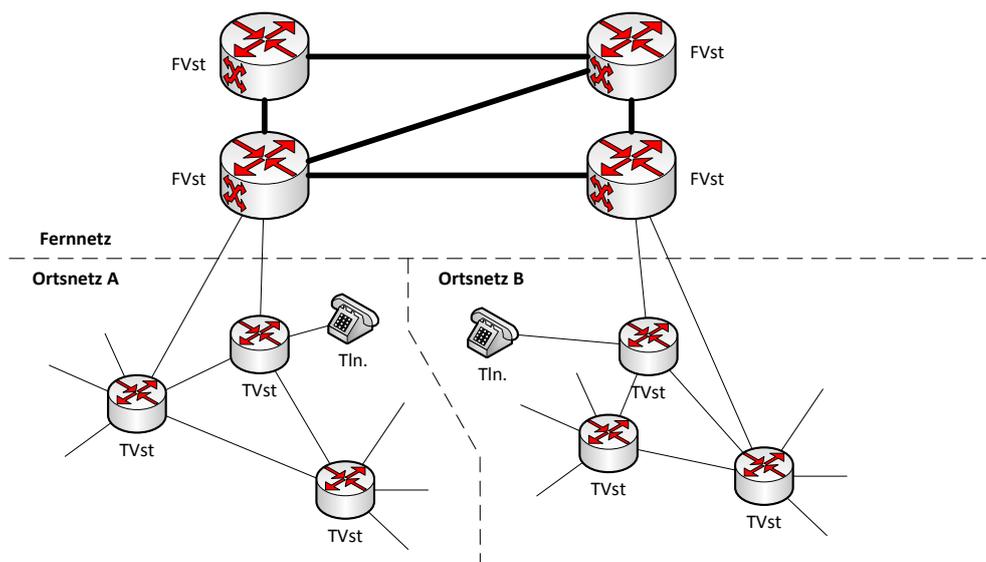


Abbildung 2.2: Netzkonzept des ISDN-Fernnetzes, die Ortsnetze mit den Teilnehmeranschlüssen werden dabei von Teilvermittlungsstellen (TVst) und das Fernnetz von Fernvermittlungsstellen (FVst) gebildet. (Quelle: [Sieg02, S.53])

der Bereichskennzahlen weiter. Sie verbinden die verschiedenen Ortsnetze miteinander und mit dem internationalen Fernnetz.

2.1.2 Mobilfunknetz 2G (GSM)

Die Einführung von Mobilfunknetzen hat eine sehr starke Verlagerung der Kommunikation vom Festnetz in die Mobilfunknetze verursacht. Bereits zehn Jahre nach der Einführung 1991 des GSM-Netzes in Deutschland, hat die Anzahl der mobilen Endgeräte bereits die Anzahl der Festnetzanschlüsse (50 Millionen) erreicht [Sieg02, S.76]. In Entwicklungsländern ohne so stark ausgebaute Telefonnetze sind Mobiltelefone noch wesentlich wichtiger für die gesamte Kommunikation. Anfang des Jahres 2009 gab es nach Zahlen der ITU-T weltweit bereits 4 Milliarden Telefonanschlüsse, wovon 2,68 Milliarden (71%) Mobilfunkanschlüsse waren [KoMS07].

GSM bildet die Grundlage für die meisten Mobilfunknetze weltweit, war aber anfangs nur als durchgängiges, digitales, europäisches Mobilfunknetz geplant. Nach den ersten analogen Mobilfunknetzen, gilt das GSM-Netz als die zweite Generation (2G) der Mobilfunknetze. Die Standardisierung wurde 1989 von der *Groupe Spéciale Mobile* (GSM) an die ETSI (siehe Kapitel 3.1.3) übergeben. Nachdem die Standards 1991 veröffentlicht wurden, blieb die Bezeichnung GSM erhalten und wurde in *Global System for Mobile Communications* umbenannt.

Ein Mobilfunknetz ist in Funkzellen aufgeteilt, die einen bestimmten geografischen Bereich abdecken. Diese Zellen überlappen sich, um einen störungsfreien Übergang zwischen zwei benachbarten Zellen zu gewährleisten (Handover). Die Funkkanäle sind über Frequenz- und Zeitmultiplexverfahren so gestaltet, dass sich verschiedene Netze und einzelne Zellen nicht gegenseitig stören können. Die Kanäle des Funknetzes arbeiten in verschiedenen Frequenzbereichen, die in Deutschland anfangs noch als D-Netz (900MHz) und E-Netz (1800MHz) bezeichnet wurden, da zu der Zeit noch nicht alle Endgeräte in beiden Frequenzbereichen funken konnten. Weil die Netze in etlichen Ländern auch andere Funkfrequenzen nutzen (etwa 850MHz und 1900MHz), können im Rahmen der Globalisierung heute viele Endgeräte in allen Netzen kommunizieren.

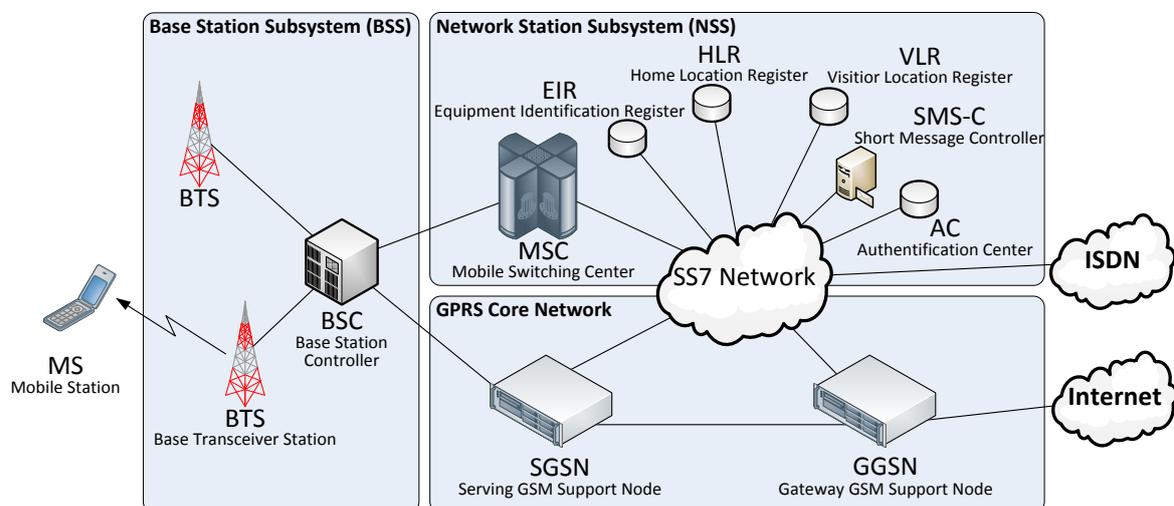


Abbildung 2.3: Das GSM-Netz besteht neben dem Zugangnetz (BSS) aus einem Network Station Subsystem (NSS), welches die Vermittlung der Telefonie und die Bewegungsfreiheit des Nutzers ermöglicht, und einem GPRS Core Network für den paketbasierten Datenzugang (frei nach [Sieg02, S.79,89]).

Eine Funkzelle besteht aus einer einzelnen *Base Transceiver Station* (BTS), welche zusammen mit andern BTS von einem *Base Station Controller* (BSC) gesteuert wird. Die zentrale Vermittlungstelle eines Mobilfunknetzes bildet das *Mobile Switching Center* (MSC). Der Übergang zum ISDN-Netz wird über ein eigenes Gateway mit einem speziellen MSC realisiert. Die Signalisierung im Mobilfunknetz basiert auf dem *Signaling System #7* (SS7). Zur Verwaltung der angemeldeten Endgeräte und der räumlichen Zuordnung in den Funkzellen existieren verschiedene Datenbanken, wie das *Home Location Register* und das *Visitor Location Register*. Eine detailliertere Ausführung dazu findet sich in [Sieg02, S.78 ff.].

Da die Zuordnung der Teilnehmer nicht mehr über eine feste Teilnehmeranschlussleitung erfolgen kann, sondern der Teilnehmer sich mit einem beliebigen mobilen Endgeräte (Mobiltelefon) am Netz anmelden kann, wird die Identifikation über ein *Subscriber Identity Module* (SIM) realisiert. Dieses Modul enthält eine interne Teilnehmerkennung (IMSI³), welcher der Netzbetreiber eine bestimmte Rufnummer in seinen Datenbanken zuordnet.

Datenübertragung im Mobilfunknetz

Da die Bandbreite eines normalen GSM-Kanals nur $9,6kBit/s$ beträgt, wurden zur Bereitstellung von höheren Bandbreiten verschiedene Techniken zur Kanalbündelung eingeführt. Im GSM-Netz hat sich dabei das *General Packet Radio System* (GPRS) durchgesetzt, welches gleichzeitig eine paketbasierende Schnittstelle für Endgeräte bietet, ohne dass die Struktur der GSM-Funkkanäle zur Sprachkommunikation verändert werden musste. Typischerweise liegt die maximale Datenrate für GPRS bei $56,6kBit/s$. Um die Datenübertragsrate nochmals zu erhöhen, wurde im Jahr 2005 *Enhanced Data Rates for GSM Evolution* (EDGE) mit einem neuen Modulationsverfahren eingeführt, was in den meisten Fällen nur ein Softwareupgrade für die BTS erforderte. Damit sind maximal $220kBit/s$ möglich. Diese Verfahren werden auch als 2.5 Mobilfunkgeneration bezeichnet (2.5G)

2.1.3 Mobilfunknetz 3G (UMTS)

Da im GSM-Netz vergleichbar mit dem schmalbandigen ISDN-Netz der Bedarf nach Anschlüssen mit einer hohen Bandbreite entstand, wurde 2004 mit dem *Universal Mobile Telecommunications System* (UMTS) die dritte Generation (3G) der Mobilfunknetze eingeführt. Das erste UMTS-Release 99 unterscheidet sich nur durch die Zugangstechnik, also neue Modulationsverfahren und Frequenzen, von den GSM-Netzen. Die interne Struktur des Netzes wurde erst mit dem UMTS Release 4 geändert, da dort kein monolithisches MSC mehr eingesetzt wird, sondern eine Trennung von Nutzdaten-transport und der Verbindungssteuerung erfolgt. Damit entspricht das UMTS-Netz nach Release 4 schon zum Teil den Standards des Next Generation Networks (siehe Kapitel 3.1.2).

UMTS bietet eine deutlich höhere Datenübertragungsrate als das GSM-Netz. Im einfachen Fall ohne Erweiterungen sind $384kBit/s$ möglich. Mit der Erweiterung *High Speed Downlink Packet Access* (HSDPA) sind durch neue Modulationsverfahren Da-

³International Mobile Subscriber Identity (IMSI)

tenraten bis zu $7,2\text{MBit/s}$ im Downlink und mit *High Speed Uplink Packet Access* (HSUPA) bis zu $5,8\text{MBit/s}$ im Uplink möglich. Durch die Einführung von größeren

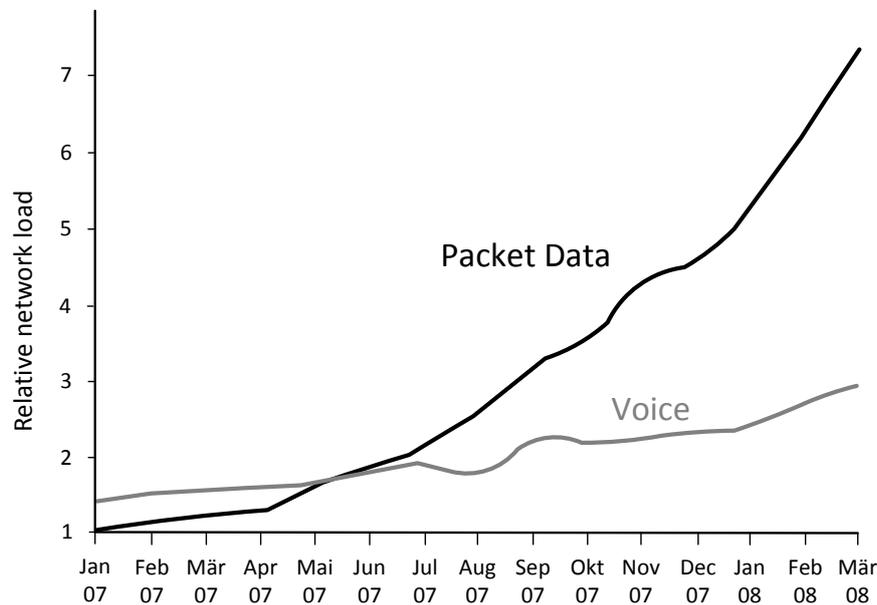


Abbildung 2.4: Seit 2007 ist der Anteil der paketvermittelten Daten in Mobilfunknetzen größer als der der Sprachdaten (*Quelle: [Opit09]*)

Bandbreiten, und damit von günstigeren Datentarifen ist der Datenverkehr in den Mobilfunknetzen ähnlich schnell gewachsen wie in den Festnetzen, und stellt seit Mitte des Jahres 2007 den größeren Anteil am gesamten Verkehrsaufkommen in den Mobilfunknetzen (Abb. 2.4) dar [Opit09].

Die nächste Generation (4G) mit noch höheren Bandbreiten, *Long Term Evolution* (LTE), befindet sich bereits in der Standardisierungs- und Erprobungsphase.

2.2 Intelligentes Netz - Dienste im ISDN

Mit der Einführung von ISDN waren erstmals Dienste im Telefonnetz möglich, die über den Übermittlungsdienst für Sprache hinausgingen [Sieg01, S.25ff]. Diese Dienste wurden unter dem Schlagwort *Intelligentes Netz* (IN) zusammengefasst, da man davon ausgegangen ist, dass diese Entwicklung das Telefonnetz in das 21. Jahrhundert bringen würde. Der fortwährend steigende Bandbreitenbedarf hat diese Planung jedoch sehr schnell überholt, so dass parallel zum Telefonnetz breitbandige Datennetze entstanden sind (siehe Kap. 2.3).

Das IN basiert auf dem architektonischen Ansatz, die Dienste in *Intelligenten Netzknoten* so zu bündeln, so dass sie nicht in jeder Vermittlungsstelle implementiert wer-

den müssen. Eine wirkliche Intelligenz gab es im Netz natürlich nicht. Der Dienst wird durch die Eingabe einer Rufnummer (Dienstkennzahl) von einem Server bereitgestellt.

```

0900          12345678
Dienstkennzahl  Dienstkunde
    
```

Die Bereitstellung der Dienste erfolgt über die im Signalisierungsprotokoll SS7 bereitgestellten Schnittstellen, wie beispielsweise der *Intelligent Network Application Part* (INAP). Für die Mobilfunknetze wurden diese Standards und Schnittstellen zu *Customised Applications for Mobile networks Enhanced Logic* (CAMEL) erweitert, die beispielsweise auch eine Abrechnung von Prepaid-Angeboten in den Netzen fremder Provider ermöglichen. Die Schnittstelle für Dienste ist hier mit *CAMEL Application Part* (CAP) bezeichnet. Zur Konfiguration der Dienste existiert ein standardisiertes Protokoll, der *Operations, Maintenance and Administration Part* (OMAP), welcher die Pflege und Verwaltung von Rufnummernblöcken oder die Festlegung von Kommunikationsziel über eine Reihe von Parametern ermöglicht.

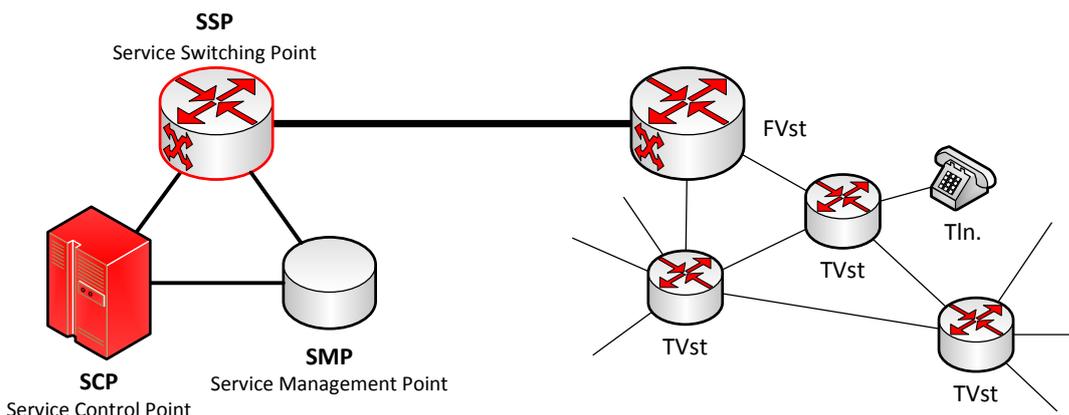


Abbildung 2.5: Dienste im Intelligenten Netz werden von einem intelligenten Netzknoten erzeugt und müssen nicht in allen Vermittlungsstellen bekannt sein.

2.2.1 Klassifikation der Dienste

Die Architektur des IN sieht vor, dass ein sehr leistungsfähiger Rechner in einem intelligenten Netzknoten die Signalisierung eines Teilnehmers (Ursprungsrufnummer, Dienstkennzahl, Tageszeit/Datum etc.) mit den Datenbankeinträgen des Dienstkunden verknüpft und entsprechend weitervermittelt. Die möglichen Dienste lassen sich in verschiedene Dienstklassen unterteilen (Abb. 2.6), die sich hauptsächlich nach der Art der Routingentscheidung oder der Funktion des Dienstes unterscheiden lassen (Abb. 2.6) [Sieg01, S. 213].

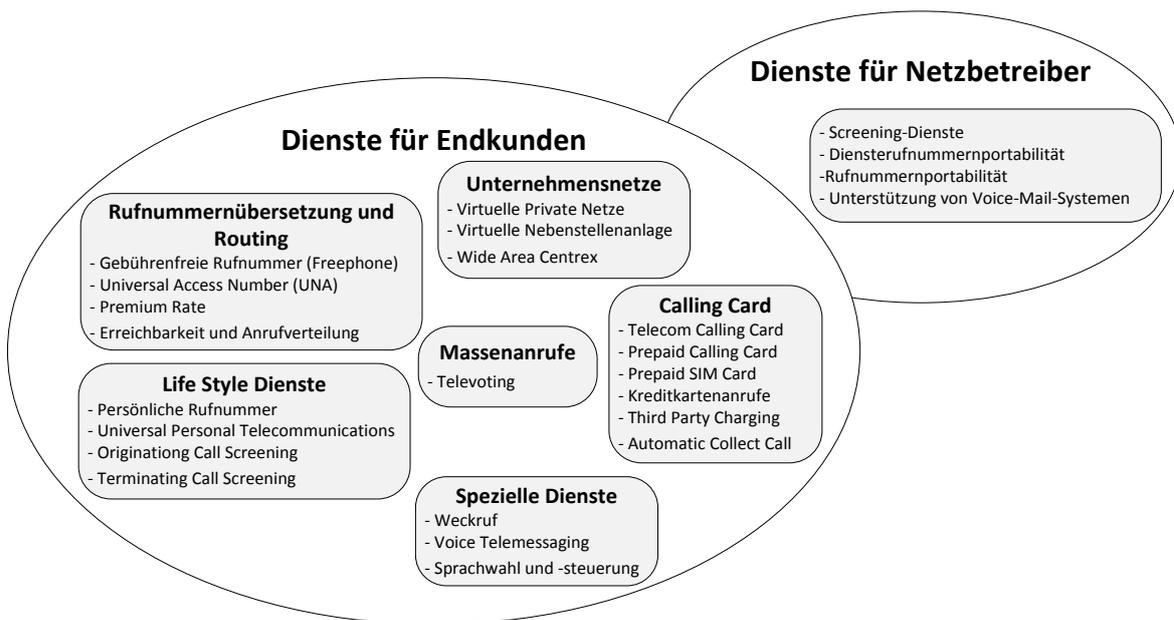


Abbildung 2.6: Dienstklassen im Intelligenten Netz (Quelle: [Sieg01, S.213])

Rufnummernübersetzung/Routing Als Techniken für routingbasierende Dienste können einfache Rufnummernübersetzung (von Dienstkundennummer auf die Nummer eines/mehrerer Anschlüsse), ursprungsabhängige Zielansteuerung, Zeit- und Datumsabhängige Zielansteuerung, Warteschleifen, Anklopfen, automatische Anrufweiterleitung oder automatisches Abweisen zum Einsatz kommen. Für die verschiedenen Dienste können auch unterschiedliche Tarife angewendet werden.

Dienste, die darauf basieren, sind etwa Freephone (Dienstekennzahl 0800 oder 0130) oder Mehrwertrufnummern für Servicedienste (0900, 0190 oder 0180). Personenbezogene Rufnummern (0700) basieren nur zum Teil auf diesen Techniken, weil hier dem Nutzer noch eine gewisse Mobilität möglich ist (Fixed Mobility), und er den Service an verschiedenen Endgeräten nutzen kann.

Mögliche Dienstleistungen, die auf diesen Techniken aufbauen, sind kostenlose und kostenpflichtige Servicehotlines, Beratungsdienstleistungen, die direkt über den Telekommunikationsanbieter über die Telefonrechnung abgerechnet werden, bis hin zu Inkassodienstleistungen. Der Notruf 112 ist ebenfalls eine solche Anwendung, da der Nutzer hier von jedem Anschluss aus eine einheitliche Servicenummer wählen kann und automatisch mit der lokal zuständigen Rettungsleitstelle verbunden wird.

Virtuelles Privates Netzwerk Das digitale Telefonnetz bietet die Möglichkeit ein virtuelles privates Netz (VPN) innerhalb des öffentlichen Telefonnetzes zu erzeugen. Das

virtuelle private Netz funktioniert für die Nutzer so, als wären sie an einer gemeinsamen Telefonanlage angeschlossen. Sie können miteinander über interne Kurzwahlen telefonieren und die internen Gespräche sind über einen Rahmenvertrag mit dem Telekommunikationsanbieter enthalten. Dieses Netz kann wiederum durch die anderen Leistungsmerkmale, wie die Rufnummernübersetzung, erweitert werden. Denkbar ist etwa eine einheitliche Nummer für die Personalabteilung einer Firma, wobei ein Mitarbeiter je nach Ursprungsnummer mit dem für ihn zuständigen Bearbeiter verbunden wird.

Calling Card Calling Cards bieten ein Abrechnungsmodell, welches nutzerbezogen und nicht mehr anschlussbezogen ist. Dazu verwendet der Nutzer zur Initiierung des Gespräches eine kostenlose Servicenummer und authentifiziert sich anschließend per Nutzernummer und Geheimzahl. Als Abrechnungsmodell kommen Prepaid- (Guthaben) oder Postpaid-Verfahren (Rechnung) zum Einsatz. Vor dem Gesprächsaufbau wird dabei der Kredit- bzw. das Guthaben des Nutzers geprüft. Vergleichbar funktionieren Kreditkartenanrufe, wobei die Abrechnung über Kreditkartenunternehmen wie VISA oder MasterCard erfolgt, und Prepaid-Karten für das GSM-Netz, die allerdings wieder über ein SIM Anschlussgebunden sind.

Dienste für Netzbetreiber Für Netzbetreiber sind mit der Einführung des digitalen Netzes ebenfalls neue Dienstleistungen möglich. Das Netz ermöglicht einen *Indirect Access*, also den Zugang zu dem Netz eines Netzbetreibers, ohne dass man eine Anschlussleitung von diesem Anbieter nutzt. Diese Möglichkeit wird für Preselect- und Call-by-Call-Angebote genutzt. Die meisten Internetzugänge per Modem oder ISDN basieren ebenfalls auf dieser Technik. Durch die Verwendung von Datenbanken für die Zuordnung und das Routing von Rufnummern ist nun auch eine Portabilität von Rufnummern möglich. Diese können zwischen verschiedenen geografischen Orten (von einem Teil eines Ortsnetzes in einen anderen), zwischen unterschiedlichen Dienstleistungen (z.B. von einem ISDN-Anschluss zu einem Analog-Anschluss) oder zwischen verschiedenen Anbietern portiert werden.

2.3 Digitale Breitbandnetze

Das ISDN-Netz war durch die begrenzte Bandbreite von $64kBit/s$ pro Kanal sehr schnell nicht mehr den steigenden Anforderungen eines schnellen Zugangs zum globalen Datennetz (Internet) für Endnutzer gewachsen.

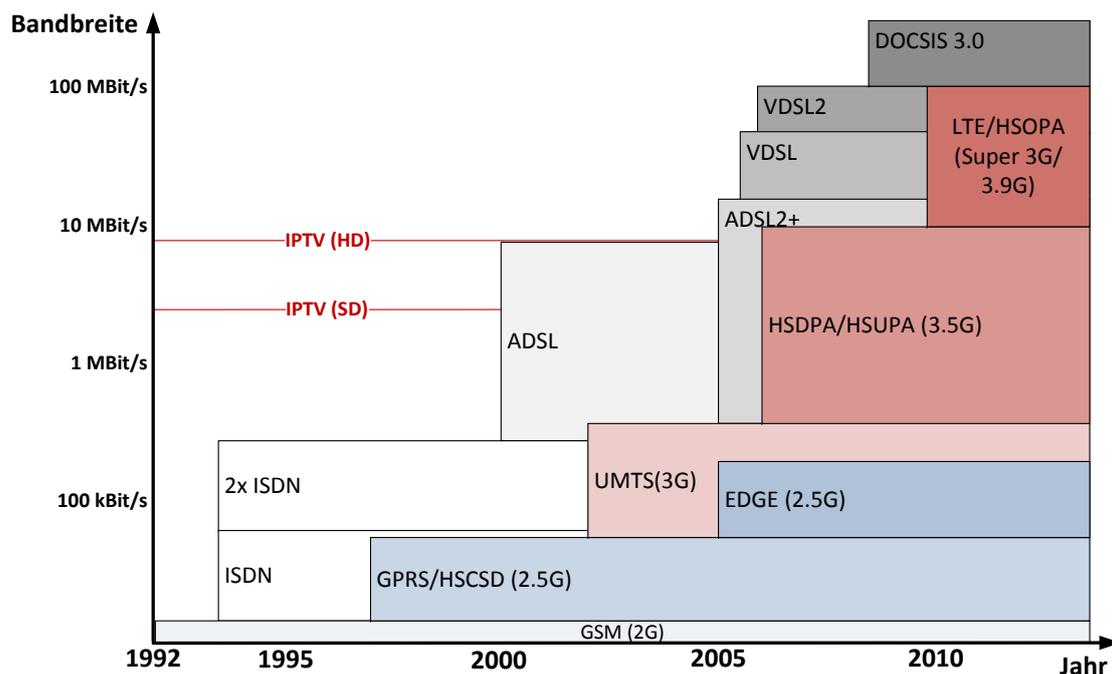


Abbildung 2.7: Breitbandstandards für Zugangsnetze und die von ihnen theoretisch bereitgestellten Bandbreiten (kabelgebunden und kabellos).

Seit 2000 setzt sich in Deutschland besonders *Asymmetric Digital Subscriber Line* (ADSL) als Breitbandzugang durch. Dabei wird auf der Teilnehmeranschlussleitung der durch ISDN nicht genutzte Frequenzbereich zwischen 138kHz und 1104kHz für die Datenübertragung verwendet. Die maximal erreichbare Datenrate liegt mit dem Standard ADSL2+ mittlerweile bei 16MBit/s im Downstream und 1MBit/s im Upstream. Teilweise ist bereits der Nachfolgestandard VDSL verfügbar, womit Bandbreiten bis zu 50MBit/s im Downstream und 5MBit/s im Upstream realisierbar sind.

Ein weiterer Zugangsweg sind Kabelmodems, welche das digitale Fernsehkabelnetz als Zugangsnetz nutzen. Dabei kommt die *Data Over Cable Service Interface Specification* (DOCSIS) zum Einsatz, die in der Version 3.0 Endnutzeranschlüsse mit einer Bandbreite von 100MBit/s ermöglicht [Brie09a].

Kabelmodems können natürlich nur in Gebieten mit entsprechend ausgebauten digitalen Fernsehnetzen genutzt werden. Die ADSL-Technologie ist größtenteils flächendeckend verfügbar, allerdings sind besonders in ländlichen Regionen durch lange Leitungen zwischen der Vermittlungstelle und der Teilnehmeranschlussleitung und teilweise niedrigen Leitungsquerschnitte und der daraus resultierenden Dämpfung der hohen Frequenzen häufig nur stark eingeschränkte Anschlüsse möglich.

2.4 Motivation für Netz der nächste Generation

Die starren Strukturen und die beschränkte Bandbreite in den bestehenden Telekommunikationsnetzen beschränken die Möglichkeiten zur Erzeugung von innovativen Diensten. Der ständig wachsende Bandbreitenbedarf im Bereich der Endnutzeranschlüsse und die große Verbreitung von Mobilfunknetzen hat mehrere getrennte Geschäftsbereiche hervorgebracht, die jeweils komplett eigene Netzwerke, Zugangsmöglichkeiten, Abrechnungssystem und Steuerungssysteme aufgebaut haben. Die Trennung dieser Netze verursacht einen hohen organisatorischen und finanziellen Aufwand. Die Nutzung der bisherigen Netze als Zugangsnetze für ein neues Next Generation Network bietet die Möglichkeit netzübergreifender Dienstleistungen, einer deutlich transparenteren Steuerung und einer einfacheren Verwaltung und Abrechnung.

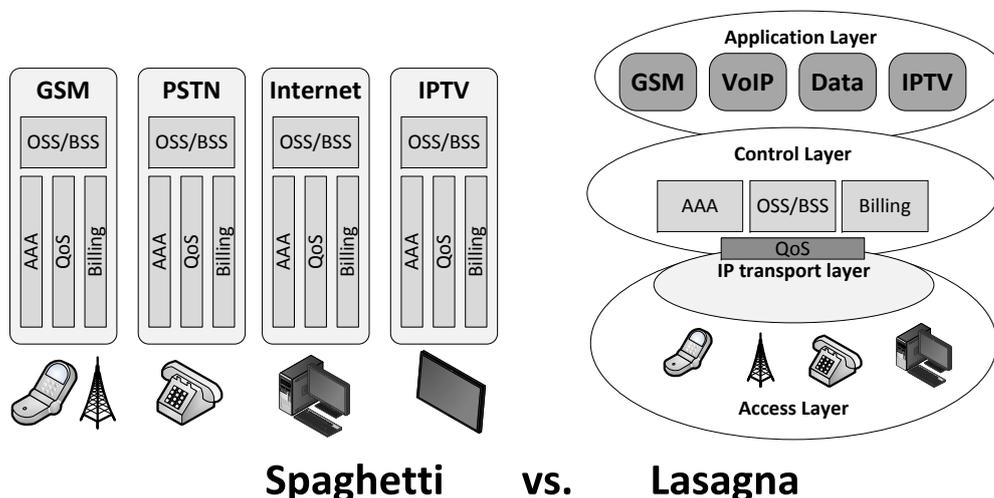


Abbildung 2.8: Konzept der herkömmlichen heterogenen Netze (links), die alle relevanten Betriebseinheiten (Abrechnung, Authentifizierung, Steuerung) separat vorhalten müssen, im Vergleich mit dem NGN, welches diese Netze als Zugangsnetze in einem mehrschichtigem Netz mit einheitlicher Steuerung vereint (Quelle: [Drlj09]).

3 Spezifikationen für Next Generation Networks

Der Transport von Sprachdaten über paketbasierende Datennetze ist schon länger Gegenstand der Forschung. Der Begriff „*Next Generation Network*“ (NGN) wird zwar von vielen Institutionen und in der Fachliteratur genutzt, aber es existiert vergleichbar mit „*Web 2.0*“ keine exakte Definition.

Die derzeit wichtigsten Neuerungen bei den Telekommunikationsnetzen laufen unter dem Stichwort „Next Generation Networks (NGN)“. Hierunter verbirgt sich ein nirgendwo exakt definiertes Konzept, das aber trotzdem relativ präzise definiert werden kann. [Tric03]

Als Beginn der Spezifizierung des Next Generation Network gilt der *NGN Workshop* der ITU-T im Juli 2003 welcher die Entwicklung IP-basierender Telekommunikationsnetzwerke unter diesem Begriff zusammenfasste [MIKO⁺07]. Daraus entstand die erste Veröffentlichung [Y2001] die den Begriff Next Generation Network erstmals genauer definierte (siehe Kapitel 3.1.1).

Direkt im Anschluss daran wurde ebenfalls im Juli 2003 von der ETSI das Projekt TISPAN gestartet. Damit begann die breite Untersuchung des NGN als Telekommunikationsinfrastruktur der nächsten Generation. Die 3rd Generation Partnership (3GPP) entwickelte mit dem IP Multimedia Subsystem (IMS) aus dem UMTS-Standard heraus parallel eine technische Basis für paketbasierende Mobilfunknetze, welche von der ETSI später für alle Netze übernommen wurde.

Die Gremien, welche maßgeblich an der Standardisierung der Eigenschaften und Funktionen von NGNs beteiligt sind sollen, auf den folgenden Seiten mit ihrer Herangehensweise vorgestellt werden. Anschließend werden die technischen Grundlagen und die Fähigkeiten des NGN erläutert.

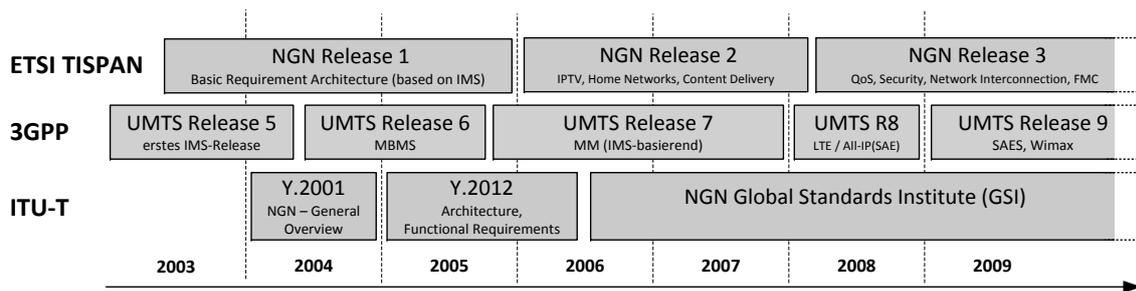


Abbildung 3.1: Veröffentlichungen der an der NGN-Spezifizierung beteiligten Gremien

3.1 Übersicht über Standardisierungsbemühungen

3.1.1 ITU - Mobilität in NGNs

Die Internationale Fernmeldeunion (kurz *ITU* für *International Telecommunication Union*) ist eine Sonderorganisation der Vereinten Nationen (UN), die sich weltweit mit den technischen Grundlagen der Telekommunikation beschäftigt. Die ITU-T hat dabei innerhalb der ITU die Aufgabe der Standardisierung der Telekommunikation. Die Standards der ITU werden als Empfehlungen veröffentlicht und von den nationalen Normungsgremien wie der ETSI oder den Regulierungsbegörden zu verbindlichen Normen umgesetzt.

Bei der ITU-T wurde die Forschung durch die 2004 neu gegründete Arbeitsgruppe *Focus Group in Next Generation Networks* (FGNGN, [ITU06]) vorangetrieben. Diese bereitete bis zum Juli 2006 verschiedene Entwürfe für die Standardisierung des NGN vor und ging Mitte 2006 in die *NGN Global Standardization Initiative* (NGN-GSI, [ITU09]) über.

Bis Oktober 2006 war der grundlegende Standardisierungsprozess abgeschlossen worden. Das Ergebnis war das „*NGN release 1 scope*“ - ein paketbasierendes Kommunikationsnetzwerk für die sitzungorientierte Multimediakommunikation zwischen zwei Teilnehmern. Dabei wurden zwar schon viele zukünftige Leistungsmerkmale und Dienste vorgestellt, der Schwerpunkt lag aber auf der sitzungorientierten Kommunikation mit Audio und Video und der Einbindung und der Emulation von ISDN-Diensten [MIKO⁺07]. Das Ziel war dabei, dass der Nutzer bei der Nutzung seines Kommunikationszugangs den Übergang zwischen NGN und dem PSTN nicht bemerken sollte. Das grundlegende Dokument war die ITU-T Empfehlung [Y2001], welches das NGN folgendermaßen definiert hat:

Das NGN ist ein paketbasierendes Telekommunikationsnetzwerk, welches

*verschiedene Breitband- und Quality-of-Service-Technologien nutzen kann. In diesem Netz sollen alle Dienstefunktionalitäten unabhängig von den zugrundeliegenden Transportfunktionen sein. Es soll den Nutzern uneingeschränkter Zugang zu verschiedenen Netzwerken und konkurrierenden Diensteanbietern bieten, so dass sie alle gewünschten Dienste nutzen können. Außerdem soll das Netz eine uneingeschränkte Mobilität für den Nutzer bieten, so dass er überall und zu jedem Zeitpunkt auf alle Dienstleistungen zugreifen kann.*¹

Eigenschaften

Für ein NGN wurden dabei folgende Eigenschaften festgelegt [Y2001, S.3]:

- paketbasierende Übertragung
- Trennung von Übertragung, Sitzungssteuerung und Anwendungen
- Entkopplung der Erzeugung von Dienstleistungen vom Datentransport, Bereitstellung von offenen Schnittstellen
- Unterstützung einer breiten Palette von Diensten, Anwendungen und Kontrollmechanismen
- Breitbandnetzwerke mit Ende-zu-Ende Dienstgütegarantien (Quality of Service)
- Verbindung zu herkömmlichen Netzen (z.B. ISDN) über offene Schnittstellen
- Nutzermobilität und Dienstmobilität
- Unbeschränkter Nutzerzugriff auf unterschiedliche Anbieter
- verschiedene Identifikationssysteme
- Einheitliche Dienstmerkmale für vom Nutzer als gleich wahrgenommene Dienste
- Konvergenz von Diensten zwischen Fest- und Mobilfunknetzen
- Unabhängigkeit der Dienste von der zugrundeliegenden Transporttechnik
- Unterstützung verschiedener Zugangsnetze
- Berücksichtigung aller gesetzlichen Anforderungen (z.B. Notruf, Abhörsicherheit, Datenschutz)

¹Frei übersetzt nach [Y2001, S.2]

Mobilität

Ein Kernpunkt der NGN-Spezifikation ist die Mobilität. Das bedeutet, dass der Nutzer oder andere mobile Endsysteme immer die Möglichkeit zur Kommunikation haben oder Dienste nutzen können, unabhängig von ihrer Position oder dem verwendeten Zugangnetz. Der Grad der Verfügbarkeit der Dienste hängt von verschiedenen Faktoren, wie den Eigenschaften des Zugangnetzes oder vertraglichen Vereinbarungen mit dem Diensteanbieter ab. Mobilität beinhaltet die Fähigkeit zu kommunizieren, unabhängig ob mit oder ohne dauerhafter Verbindung.

Architektur

Um die beschriebenen Eigenschaften des NGN zu erreichen, besteht das NGN-Konzept der ITU-T grundsätzlich aus zwei Schichten: einer Transportschicht und einer Service-schicht [Y2011, S.5 ff.] (siehe Abb. 3.2) .

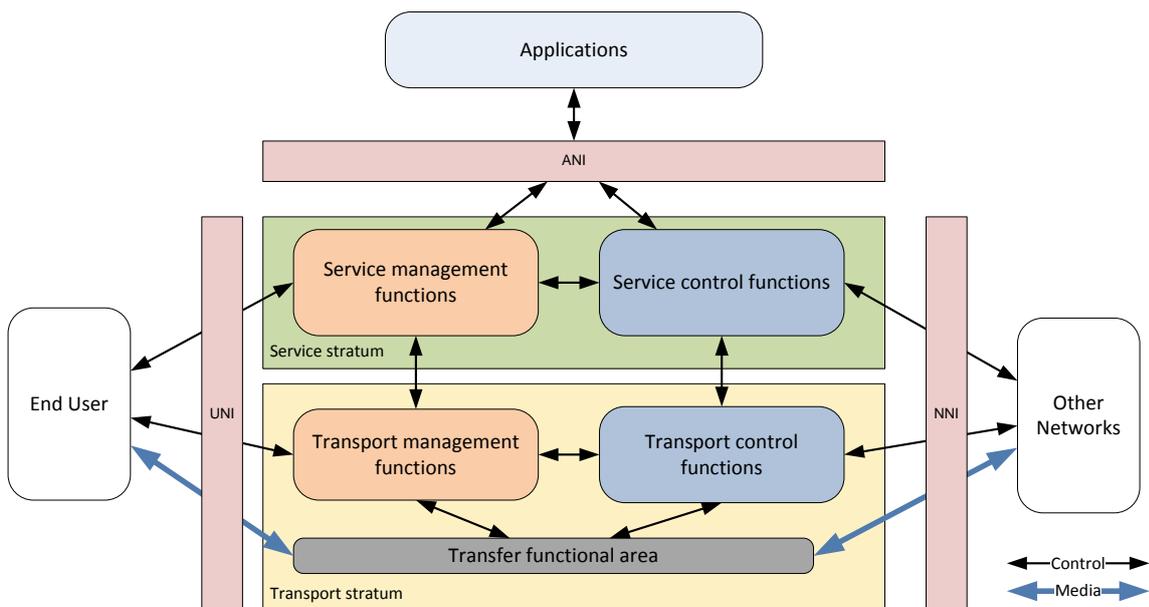


Abbildung 3.2: Das Referenzmodell des NGN der ITU besteht aus einer Service- und einer Transportschicht. Für die Bereitstellung von Anwendungen steht genauso wie für den Nutzerzugang und die Anbindung fremder Netze jeweils eine Schnittstelle bereit. (frei nach [Y2011, S.9] und [MIKO⁺07])

Die Serviceschicht beinhaltet Funktionen, die den Zugriff auf die Kommunikationsdienste ermöglichen (Benutzerauthentifizierung, Sitzungs- und Verbindungssteuerung, Aushandlung Sitzungsparameter). Die Bereitstellung der Sitzungssteuerung basiert auf dem *Session Initiation Protokoll* (SIP, siehe Kapitel 3.2.2).

Die Transportschicht ist primär für den Transport der IP-Pakete in einer Ende-zu-Ende-Verbindung mit Dienstgütegarantien und Sicherheit (Security) zuständig. Dabei sind Kontrollfunktionen (Management and Control Functions) vorgesehen, welche die Einhaltung von Servicegarantien ermöglichen sollen, die beispielsweise eine einmal zustandegekommene Verbindung dauerhaft gewährleisten sollen.

Das NGN-Konzept besitzt drei verschiedene Schnittstellen. Das *User Network Interface* (UNI) dient als Nutzerschnittstelle zum Zugriff auf das Kommunikationsnetzwerk, das *Network Network Interface* (NNI) zur Verbindung mit anderen Netzwerken (Provider) und das *Application Network Interface* (ANI) zur Bereitstellung von weiteren Diensten und Anwendungen. Die Schnittstellen werden in [Y2012] detailliert spezifiziert.

3.1.2 3GPP - NGNs auf Basis von Mobilfunknetzen

3GPP (kurz für *3rd Generation Partnership Project*) stellt eine weltweite Kooperation von Standardisierungsgremien im Mobilfunk dar. Dazu gehören unter anderem die ETSI, ATIS (USA), TTC (Japan) und ein Großteil der Mobilfunknetzbetreiber und Netzausrüster wie Ericsson oder Nokia Siemens.

Die 3GPP hat mit der Veröffentlichung von *UMTS Release 5* 2004 das erste Mal ein reines IP-basierendes Netzwerk für die Mobilkommunikation in UMTS und den Nachfolgenetzen vorgesehen. In UMTS Release 4 wurde zwar schon eine Trennung von Nutzdatentransport und Signalisierung vorgesehen (siehe Kapitel 2.1.3), was aber ohne paketbasierendem Nachrichtentransport noch nicht vollständig dem Grundgedanken des NGN entsprach.

Ursprünglich war das *IP Multimedia Subsystem* (IMS) nur für die Verbreitung von Internetdiensten und Sprachtelefonie über GPRS und UMTS vorgesehen. Da das IMS aber nicht zwingend eine bestimmte Zugangstechnologie voraussetzt, wurde es von ETSI für ihr *NGN Release 1* als Grundlage verwendet. Anschließend wurde in späteren Veröffentlichungen der 3GPP IMS auch auf andere drahtlose Netzwerke (WLAN, WiMax) und leitungsgebundene Netzwerke (xDSL, Fernseekabel) ausgedehnt, und bildet so die Grundlage für ein einheitliches Kernnetz mit Mobilität zwischen den Zugangnetzen.

IMS - IP Multimedia Subsystem

Das Netzmodell des IMS wird in [3GPP23228] beschrieben und besteht aus 3 Schichten, welche die Trennung von Datentransport und den Leistungsmerkmalen des Netzes und

den Anwendungen sicherstellen.

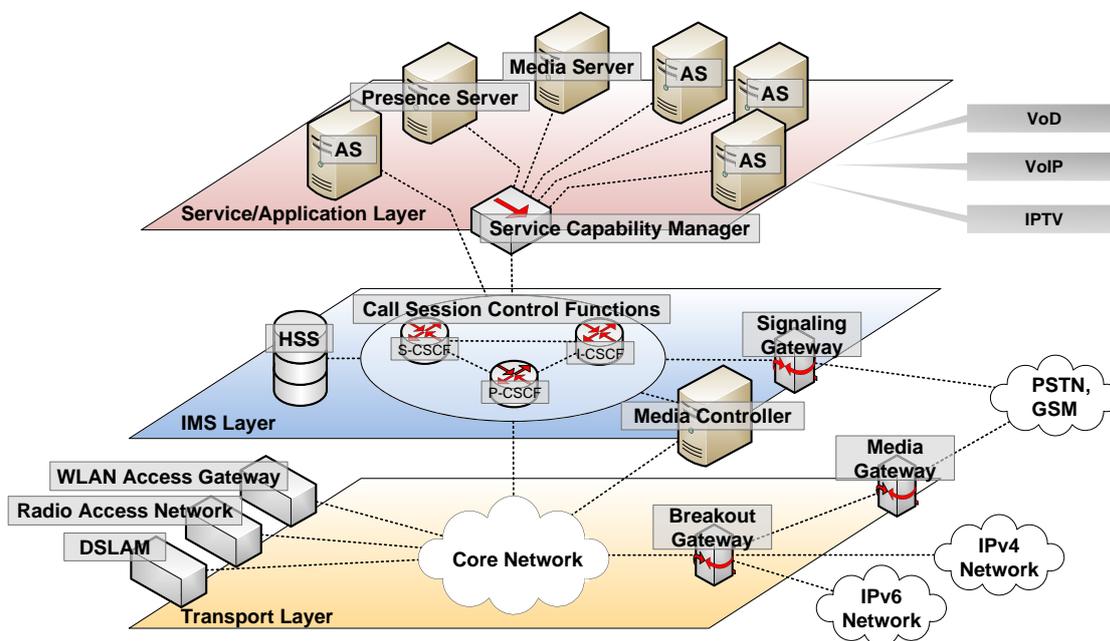


Abbildung 3.3: Das IP Multimedia Subsystem (IMS) besteht aus drei Schichten, welche die Trennung von Datentransport, Sitzungs- und Verbindungssteuerung (CSCF) und Anwendungen sicherstellen (frei nach [Siet09, S.93] und [DrIj09]).

Die Transportschicht beinhaltet das Kernnetz, ein IP-basierendes Transportnetz mit QoS-Funktionen (siehe Kapitel 3.5.1). An das Kernnetz sind die einzelnen Zugangnetze angebunden, über welche der Endnutzer auf das NGN zugreifen kann. Die IMS-Schicht beinhaltet alle Funktionen zur Sitzungssteuerung und Signalisierung von Verbindungen via SIP (siehe Kapitel 3.2.2). Diese Elemente werden als *Call Session Control Functions* (CSCF) bezeichnet. Die Anbindung an das PSTN und GSM erfolgt über Gateways (siehe Kapitel 3.3.5). Die Kommunikationsdienste, auf welche der Nutzer letztendlich zugreift, sind ein Teil der Service- und Anwendungsschicht. Von den Anwendungsservern in dieser Schicht werden dann die einzelnen Dienste, wie Telefonie (VoIP), Fernsehen (IPTV) oder Video-on-Demand (VoD) bereitgestellt. Dabei ist auch ein *Service Capability Manager* vorgesehen, der die Anfrage nach einem bestimmten Dienst ohne direkte Adressierung eines Endsystems an einen Anwendungsserver, der diesen Dienst anbietet, weiterleitet.

3.1.3 ETSI TISPAN - NGNs als Ersatz für ISDN

Das Europäische Institut für Telekommunikationsnormen (kurz *ETSI* für *European Telecommunications Standards Institute*) gehört zu den drei großen Normungsorganisationen Europas und wurde 1988 auf Initiative der Europäischen Union gegründet. Das ETSI ist für die Normung im Bereich der Telekommunikation zuständig, und hat beispielsweise an den Standards für ISDN (DSS1) (siehe Kapitel 2.1.1) und GSM mitgewirkt. Beide System waren ursprünglich nur für die bessere Vernetzung der europäischen Länder gedacht, kamen aber aufgrund des großen Erfolges weltweit zum Einsatz. GSM wird heute beispielsweise in 200 Ländern genutzt.

Aus der Zusammenlegung von den Projekten *ETSI TIPHON* (Telecommunications and Internet Protocol Harmonization Over Networks) und *ETSI TC SPAN* (Technical Committee Services and Protocols for Advanced Networks) wurde *ETSI TC TISPAN* (Telecoms & Internet converged Services & Protocols for Advanced Networks). Diese Einheit verwaltet heute die Standards für NGNs innerhalb der ETSI und besteht aus acht Arbeitsgruppen, welche die einzelnen Teilbereiche der Spezifizierung betreuen (Abb. 3.4 links) [KuHi06, S.53,56].

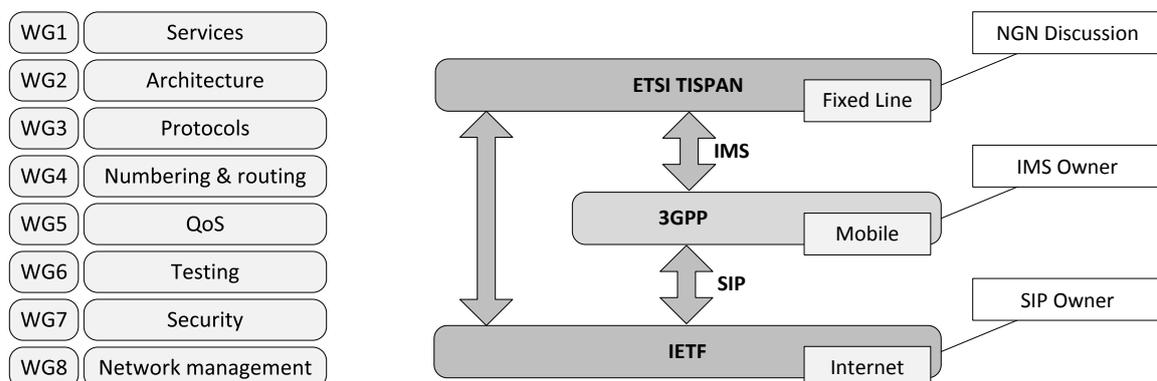


Abbildung 3.4: Arbeitsgruppen zur Spezifizierung des NGN innerhalb der ETSI (links); Organisationen, die an der NGN-Spezifikation des ETSI beteiligt sind (Quelle: [KuHi06])

Die NGN-Spezifikation des ETSI basiert auf dem IMS (siehe Kapitel 3.1.2) der 3GPP. Das IMS wurde zur Verwendung im NGN über kabelgebundene Netzwerke erweitert, um eine gemeinsame technische Basis zu erhalten und die bestehenden Kommunikationsnetze einzubinden. Die für die Sitzungssteuerung im IMS und die Emulation des PSTN verwendeten Protokolle (SIP, RTP) stammen wiederum von der *Internet Engineering Task Force* (IETF) (Abb. 3.4, rechts). Das *NGN Release 1* [ETSI181001] wurde im Dezember 2005 veröffentlicht (Übersicht in [Tech09]). Ziel von Release 1 war einen

Migrationspfad zum Ersetzen von PSTN/ISDN zu erhalten (Emulation Service) und Multimediaverbindungen in drahtgebundenen Netzwerken zu ermöglichen. Die Emulation der Legacy-Dienste (PSTN/ISDN) soll für Benutzer transparent sein, so dass beispielsweise vorhandene Terminals via Adapter problemlos mit vollem Leistungsumfang weiterverwendet werden können. In den folgenden Releases wurden jeweils weitere Funktionen und Dienste hinzugefügt.

Aufbau

Das Kernnetz, welches die *Call and Session Control Functions* (CSCF) wie Signalisierung, Routing und Nutzeranmeldung bereitstellt, wurde im Zuge der IMS-Einführung von der 3GPP übernommen. Das 3-schichtige NGN-Modell der ETSI baut deswegen auf dem Modell der 3GPP auf und umfasst die gleichen Elemente (Abb. 3.5). Das NGN war beim ETSI allerdings von Anfang an darauf ausgelegt, dass als Zugangsnetze alle verfügbaren drahtlosen und drahtgebundenen Technologien eingesetzt werden können. Im Konzept der 3GPP wurde das erst nach und nach in UMTS Release 6 und 7 aufgenommen.

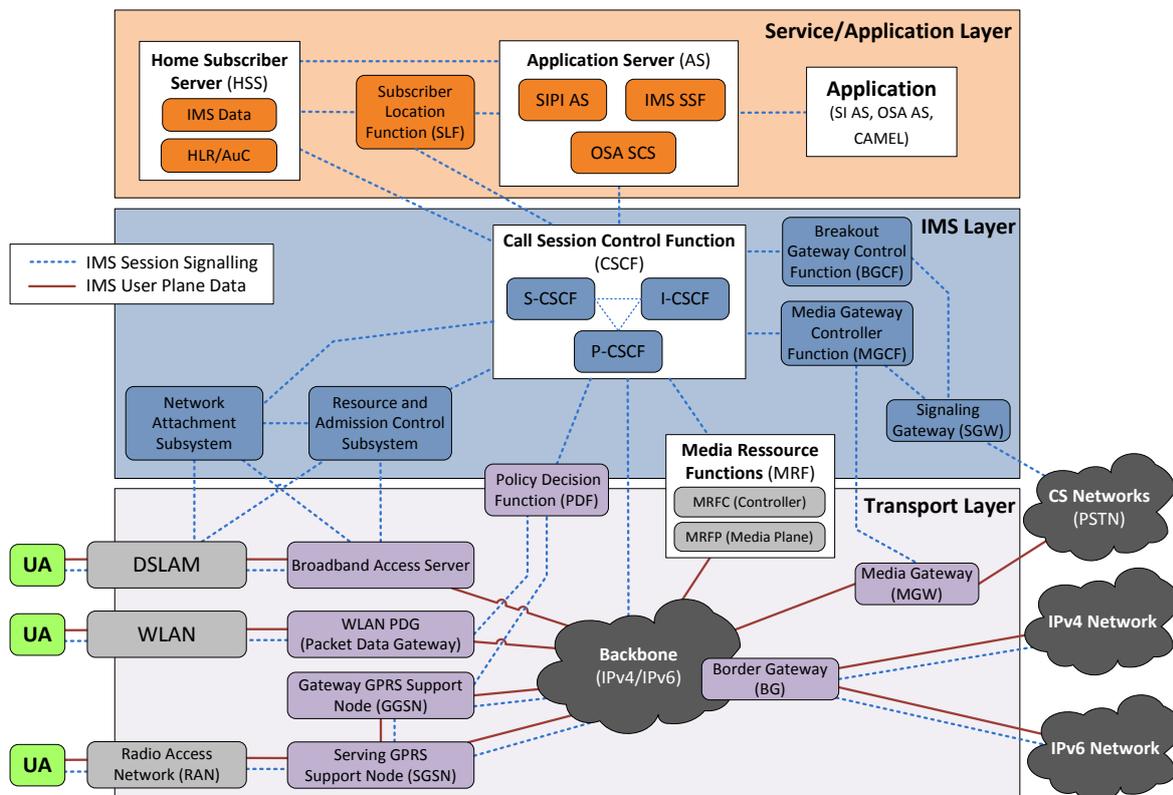


Abbildung 3.5: Das IP Multimedia Subsystem (IMS) der 3GPP bildet den Kern des ETSI NGN-Modells (nach [Siet09, S.93]).

Den Kern der Transportschicht bildet hier ein ein IP-basierendes Netzwerk, welches über ein *Border Gateway* (BD) mit anderen IP-Netzen verbunden ist. Der Nutzerzugang erfolgt über verschiedene Zugangsnetze, wie beispielsweise DSL, WLAN oder Funknetze (GSM, UMTS, WiMAX). Die Sitzungs- und Verbindungssteuerung in der IMS-Schicht erfolgt über die CSCF, wobei hier in *Proxy-* (P-CSCF), *Serving-* (S-CSCF) und *Interrogating-CSCF* (I-CSCF) unterschieden wird. Eine Anbindung anderer leitungsvermittelnder Kernnetze (*CS Networks*) erfolgt über *Signalisierungs-* (SGW) und *Mediengateways* (MGW). Die Konvertierung inkompatibler Audioströme oder die Bereitstellung von Konferenzen wird von den *Media Resource Functions* (MRF) realisiert. In der Anwendungsschicht sind die Elemente angesiedelt, welche zur Bereitstellung von Diensten (*Anwendungsserver*, AS) und zur Realisierung von Mobilität (*Subscriber Location Function*, SLF und *Home Subscriber Service*, HSS) nötig sind. Die einzelnen Netzelemente werden mit ihren Funktionen in Kapitel 3.3 beschrieben.

3.1.4 Zusammenfassung

Die NGN-Konzepte der ITU-T, ETSI und 3GPP unterscheiden sich insgesamt nur in Details, wie die Bezeichnung und die Zusammenfassung einzelner Elemente. Die Service- und Anwendungsschicht, die in den IMS-basierenden Modellen vorhanden ist, wird im ITU-Modell jedoch nur durch eine Schnittstelle zu Anwendungsservern definiert.

Die verwendeten Netzelemente sind, besonders in der Vermittlungs- und Steuerungsschicht und in der Transportschicht, bei allen Konzepten sehr ähnlich. Alle Konzepte nutzen mittlerweile SIP zur Signalisierung und RTP für den Datentransport. Die ITU-T hat anfangs ihren eigenen Protokollstack [H323] favorisiert. Dieser ist aber mittlerweile nur noch ein optionales Leistungsmerkmal.

Zunächst werden die technischen Grundlagen der Sitzungs- und Verbindungssteuerung sowie des Nutzdatentransports betrachtet. Anschließend werden die einzelnen Netzelemente des NGNs und die Hauptmerkmale beschrieben.

3.2 Technische Grundlagen

3.2.1 Netzwerkprotokolle

3.2.1.1 Internet Protocol (IP)

Die Kommunikation im NGN soll, wie in den letzten Abschnitten beschrieben, über ein paketbasierendes Netzwerk ablaufen. Da das *Internet Protokoll* (IP) mittlerweile das Standardprotokoll für die paketbasierende Kommunikation in Datennetzen ist, wird in der Regel davon ausgegangen, dass es auch die Grundlage für die Kommunikation im NGN bilden wird.

Das Internet Protokoll ist für die Vermittlung der einzelnen Datenpakete zwischen den Knoten (Routern) im Netzwerk zuständig. Es ist unabhängig von den darunterliegenden Sicherungs- und Bitübertragungsschichten, so dass dort verschiedenste Technologien wie Ethernet, Token-Ring oder ADSL/ATM mit PPP zum Einsatz kommen können. Für den Datentransport werden in IP-Netzwerken grundsätzlich zwei Netzwerkprotokolle verwendet: TCP und UDP.

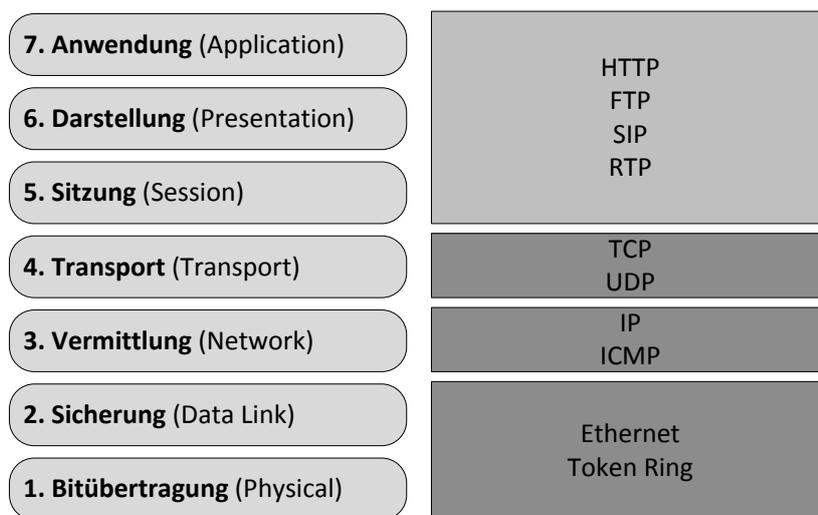


Abbildung 3.6: Schichten des OSI-Referenzmodells in einem IP-Netzwerk, mit Beispielen für Übertragungsprotokolle und -techniken

3.2.1.2 Transmission Control Protocol (TCP)

Für die meisten Internetanwendungen (HTTP, E-Mail, FTP etc.) wird das *Transmission Control Protocol* (TCP) genutzt, das eine „verbindungsorientierte“ Übertragung von Paketen ermöglicht. Verbindungsorientiert bedeutet in einem paketbasierendem Netzwerk jedoch nur, dass TCP einen virtuellen Kanal zwischen zwei Endpunkten (Sockets)

herstellt. Dabei ist eine bidirektionale Datenübertragung möglich und Datenverluste und Überlastungen können durch Bestätigungspakete erkannt werden.

3.2.1.3 User Datagram Protocol (UDP)

Das *User Datagram Protocol* (UDP) ist dagegen vollständig verbindungslos und dadurch auch deutlich einfacher aufgebaut. UDP-Pakete werden einfach an einen Port eines entfernten Zielrechners gesendet und dort entsprechend weiterverarbeitet. Um Datenverluste oder Überlastungen auf dem Übertragungsweg müssen sich die Anwendungen der darüberliegenden Schichten selbstständig kümmern. Dafür lassen sich durch den geringeren Overhead höhere Datenraten und eine schnellere Verarbeitung erreichen. Da bei verlorenen Paketen nicht erst auf Bestätigungspakete gewartet werden muss, um den Datenstrom fortzusetzen, sind mit UDP auch deutlich kürzere Paketlaufzeiten erreichbar.

Da die Laufzeit der Pakete, und damit die Verzögerung von Sprach- und Videoübertragung, sowie die effektive Nutzung der Bandbreite in einem paketbasierendem Netz die größten Störfaktoren bei Echtzeitanwendungen sind, kommt für die Nutzdatenübertragung UDP zur Anwendung. Für die Signalisierung wären die verbindungsorientierten Eigenschaften von TCP von Nutzen, da sonst nicht festgestellt werden kann, ob ein Zielrechner überhaupt erreichbar ist oder das Paket unterwegs verloren gegangen ist. Die Protokolle für die Signalisierung im NGN besitzen diese Funktionalitäten jedoch selbst, so dass sie nicht auf eine TCP-Übertragung angewiesen sind.

3.2.1.4 Sitzungsprotokolle

Da die Kommunikation in einem IP-Netz aus der Sicht der Telekommunikation immer noch verbindungslos ist, müssen die Protokolle in den darüberliegenden Schichten eine Verbindungs- und Dienstesteuerung bereitstellen. Dafür kommen mehrere Protokollfamilien in Frage, wie das *Session Initiation Protocol* (SIP, entwickelt von IETF) [RFC3261], H.323 (entwickelt von ITU-T) [H323] und das *Media-Gateway-Control-Protocol* (MGCP) [RFC3435]. Das MGCP ist eigentlich für die Steuerung für VoIP-Gateways im Netz gedacht. Die beiden anderen Protokolle sind direkt für die Kommunikation bis zum Endnutzer entwickelt worden. Die Protokolle sind allerdings nur in den Grundfunktionen kompatibel, so dass effektiv nur eines der Protokolle genutzt werden kann oder die Kommunikation zwischen zwei Teilnehmern über einen Server laufen muss, der die Nachrichten entsprechend übersetzt.

Da SIP im Release 5 von UMTS als Standard festgelegt wurde und als das leis-

tungsfähigere und leichter erweiterbare Protokoll gilt, ist es mittlerweile zum einzigen Standard für Echtzeitkommunikation im NGN geworden [TrWe07, S.33].

3.2.2 SIP - Session Initiation Protokoll

Die Funktion von SIP im NGN ist vergleichbar mit einem Signalisierungsprotokoll in einem herkömmlichen Telekommunikationsnetzwerk. Die erste Spezifikation für SIP wurde im Jahr 1999 von der IETF als [RFC2543] veröffentlicht. Aktuell sind die Grundlagen und einige Erweiterungen von SIP in den RFCs [RFC3261, RFC3262, RFC3264, RFC3265] festgelegt.

SIP bietet allerdings nur Funktionalitäten für den Aufbau, die Steuerung und den Abbau einer Kommunikationssitzung zwischen mehreren Teilnehmern. Zur Übertragung der Nutzdaten wird das *Real-Time Transport Protocol* (RTP) genutzt (siehe Kapitel 3.2.5).

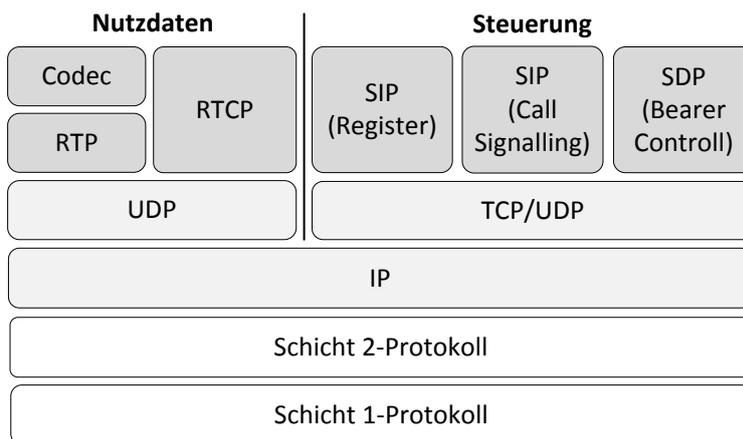


Abbildung 3.7: Protokoll-Stack für SIP-Verbindungen

Neben dem Auf- und Abbau von Kommunikationssitzungen unterstützt SIP auch den Austausch von Textnachrichten oder die Abfrage des Zustandes anderer Teilnehmer. Die SIP-Nachrichten können per UDP, TCP oder SCTP (Stream Control Transmission Protocol, [RFC2960]) transportiert werden, da SIP selbst verbindungsorientiert arbeitet und Verfahren zur Kommunikationssicherung gegen Zeitüberschreitungen oder Datenverluste mitbringt.

Die Endsysteme, welche SIP-Verbindungen auf der Nutzerseite terminieren oder einleiten, werden als *User Agent* (UA) bezeichnet. Ein solcher UA kann ein SIP-Softphone oder ein Hardware-IP-Telefon sein (siehe Kapitel 3.3.1). Die Funktion der einzelnen Systeme im Kommunikationsvorgang wird im Kapitel 3.3 genauer beschrieben.

3.2.2.1 SIP URI

Endsysteme werden bei SIP über sogenannte *Uniform Resource Identifier* (URI, teilweise auch SIP URL) adressiert. URIs sind ein Standard für die Adressierung von physikalischen Systemen und Ressourcen, der in den meisten Computernetzwerken genutzt wird [RFC3986]. Die bekanntesten Schemata die URIs verwenden sind *http* für Webserver im Internet und *mailto* für E-Mail-Adressen. Eine SIP-URI entspricht am ehesten einer E-Mail-Adresse mit einer vorangestellten Protokollbezeichnung, und hat die Form `sip:user@host`.

Der „User“-Teil der SIP-URI ist dabei eine individuelle Bezeichnung des Nutzers und der „Host“ ist eine IP-Adresse oder ein Domain-Name. Ein SIP User Agent erzeugt bei der Aktivierung immer eine temporäre SIP URI, die abhängig vom verwendeten Rechner und vom genutzten Netzwerk ist, da der „Host“ in diesem Fall die aktuelle Netzwerkadresse ist.

temporäre SIP URIs:

```
sip:alice@141.24.93.180
```

```
sip:866623@192.168.2.1
```

Beim Wechsel des Endsystems oder des Netzwerkes ändert sich also zwangsläufig die IP-Adresse des Nutzers und damit seine SIP URI. Zur direkten Kommunikation muss ein Gesprächspartner dementsprechend immer die temporäre SIP URI des anderen Nutzers kennen.

Um dieses Problem zu umgehen, existieren Anbieter mit einer vermittelnden Infrastruktur, welche den Nutzern dauerhafte URIs zuordnen können. Dazu meldet sich der Nutzer mit seinem User Agent aus einem beliebigen Netzwerk am System des Anbieters an (Registrar Server) und teilt ihm seine temporäre SIP URI mit. Diese wird im Location Server (siehe Kapitel 3.3) zusammen mit seiner dauerhaften SIP URI gespeichert. Diese dauerhafte SIP URI ist genauso aufgebaut wie die temporäre, nur das „User“ durch den Nutzernamen beim Anbieter und „Host“ durch die Domain des Anbieters ersetzt werden.

dauerhafte SIP URIs:

```
sip:bob@voyager.e-technik.tu-ilmenau.de
```

```
sip:866623@sipgate.de
```

Wünscht nun ein anderer Teilnehmer ein Gespräch mit einem Nutzer dieses Anbieters, kontaktiert er ihn auf seiner dauerhaften SIP URI. Der Location Server des Anbieters liefert die aktuell gültige temporäre SIP URI an ein vermittelndes Netzelement (z.B.

SIP Proxy Server), der das Gespräch anschließend zwischen den zwei Teilnehmern einleitet.

3.2.2.2 SIP Nachrichten

SIP nutzt zur Kommunikation prinzipiell zwei Arten von Nachrichten, *SIP-Requests* (Anfragen) und *SIP-Responses* (Statusinformationen). Im Kommunikationsvorgang hat das Endsystem, welches die Anfrage (SIP-Request) verschickt, nach dem Client-Server-Modell immer die Rolle des Clients und das antwortende System die des Servers.

Anfragen Die Basisnachrichtentypen für Anfragen bei SIP umfassen sechs grundlegende Nachrichten die für den Auf- und Abbau einer einfachen Verbindung nötig sind.

INVITE ist ein Nachrichtentyp zum Aufbau einer SIP-Sitzung zwischen zwei Endsystemen. Die Nachricht löst den Verbindungsaufbau (siehe Kaptiel 3.2.2.3) aus und enthält bereits Parameter zur Multimediasitzung. Diese Parameter werden in einem gekapselten SDP-Paket übertragen (siehe Kapitel 3.2.3). Die Nachricht muss nicht von einem der Teilnehmer gesendet werden, sondern kann auch von einem dritten System aus den Verbindungsaufbau einleiten. Durch eine INVITE-Nachricht in einer bestehenden Sitzung können die Sitzungsparameter modifiziert werden.

BYE beendet eine bestehende Sitzung.

OPTIONS ermöglicht die Abfrage von Informationen über ein Endsystem ohne eine bestehende Sitzung.

CANCEL dient zum Abbruch einer vorher gesendeten Anfrage, beispielsweise zum Beenden einer Sitzung bevor diese etabliert ist.

ACK bestätigt den Empfang einer Statusinformation beim Three-Way-Handshake (siehe nächster Abschnitt). Diese Nachricht ist keine Statusinformation, da sie im Gegensatz zu diesen alle Verbindungsdaten enthält (SIP URIs der Teilnehmer, Call-ID etc.).

REGISTER dient zur Registrierung eines User Agents bei einem Anbieter von dauerhaften SIP URIs.

Erweiterte Anfragen Es existieren außerdem erweiterte Anfragen, welche aber nicht von allen User Agents verarbeitet werden können. Diese Anfragen sind für erweiterte Funktionalitäten außerhalb des Auf- und Abbaus einer Multimediasitzung. Solche Anfragen sind beispielsweise:

SUBSCRIBE dient zur Anforderung der Zustandes eines Endsystems, und kann zur Abfrage von Präsenzinformationen (Online-Status) verwendet werden..

REFER ist die Aufforderung an einen User Agent, ein anderes Endsystem zu kontaktieren, etwa zur Einleitung eines Gespräches (Third-Party-Call, siehe Kapitel 5.1.4) oder zur Übernahme eines Gespräches.

NOTIFY wird als Antwort auf die Anfragen SUBSCRIBE oder REFER verwendet.

MESSAGE übersendet eine einfache Textnachricht an ein anderes Endsystem, beispielsweise für die Realisierung von Chats.

INFO wird zum Transport von beliebigen Steuer- und Kontrollanforderungen verwendet, die nicht auf dem SIP-Protokoll basieren. Damit wird unter anderem das Tunneln von ISDN-Steuerinformationen möglich ohne diese in SIP zu konvertieren.

Statusnachrichten Die Statusnachrichten (Responses) [RFC3261, S.28 ff.] dienen zur Bestätigung von eingegangenen Anfragen. Die Zuordnung der Statuscodes entspricht in etwa den Statusnachrichten von http. Als Statusinformationen kommen dabei folgende Klassen zum Einsatz:

1xx Vorläufige Statusinformationen über Anfragen, die noch nicht vollständig abgearbeitet sind. Beispiele hierfür sind *100 Trying* oder *180 Ringing*. Diese Informationen sind erst dann von Bedeutung, wenn die Verbindung über eine SIP-Infrastruktur hergestellt wird (siehe Abschnitt Verbindungsaufbau).

2xx Erfolgsmeldung über eine eingegangene und abgearbeitete Anfrage. Diese Klasse umfasst nur zwei Meldungen, *200 OK* und *202 Accepted*.

3xx Information über eine Umleitung, d.h. die Anfrage konnte nicht abgearbeitet werden, ohne dass weitere Schritte unternommen werden. Damit kann beispielsweise auf eine geänderte SIP URI (*302 Moved temporarily*) eines Nutzers verwiesen werden.

- 4xx** Meldung über eine fehlerhafte Anfrage, die auf eine fehlende Authorisierung (*401 Unauthorized*), eine nicht vorhandene Ressource (*404 Not found*), eine fehlerhafte Syntax (*416 Unsupported URI Scheme*) oder andere Fehlerzustände hinweist.
- 5xx** Serverbedingte Fehlermeldung, die darauf hinweist, dass der Server durch einen internen Fehler, beispielsweise eine Fehlfunktion (*500 Server internal error*) oder einen nicht erreichbaren Dienst (*503 Service unavailable*) die Anfrage nicht bearbeiten kann.
- 6xx** Globale Fehler, bedeutet dass die Anfrage zwar bis zum gewünschten Teilnehmer durchgestellt werden konnte, aber aufgrund globaler Probleme, wie Überlastung (*600 Busy everywhere*), die Anfrage nicht komplett abgearbeitet wurde.

3.2.2.3 Verbindungsaufbau

Der Verbindungsaufbau bei SIP ist durch den *Three-Way-Handshake* gekennzeichnet (siehe Abb. 3.8). Die SIP-Sitzung selbst hat das Ziel, innerhalb der Sitzung einen verbindungsorientierten Kanal zur Übertragung von Multimediatelephonedaten bereitzustellen. Der

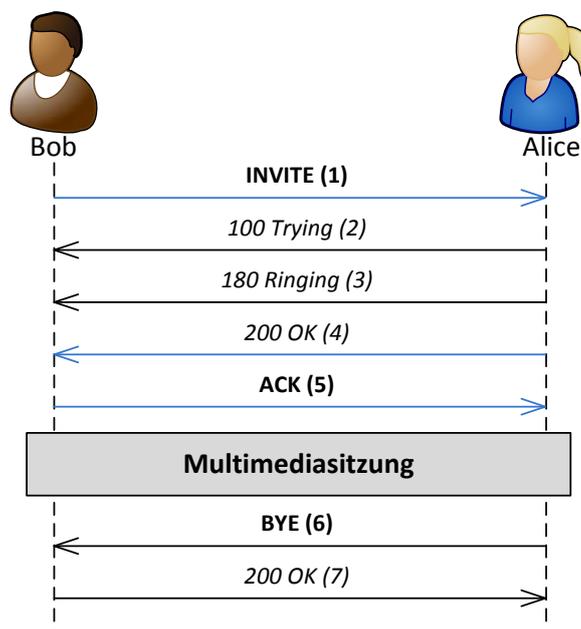


Abbildung 3.8: Three-Way-Handshake (blau gekennzeichnet) beim Aufbau einer SIP-Verbindung, wobei SIP-Anfragen (fett) und SIP-Statusnachrichten (kursiv) ausgetauscht werden.

Three-Way-Handshake wird durch eine *INVITE*-Anfrage ausgelöst. Diese kann direkt an die temporäre SIP URI eines anderen Nutzers oder an die dauerhafte SIP URI eines

Nutzers bei einem Anbieter gerichtet sein. Anschließend können vorläufige Statusinformationen übertragen werden, welche aber keinen Einfluss auf den Verbindungsaufbau und den Three-Way-Handshake haben. Beantwortet wird die INVITE-Anfrage erst durch eine finale Bestätigung durch die Statusnachricht *200 OK*. Endgültig etabliert wird die Verbindung erst durch die ACK-Anfrage des Initiators.

Der Verbindungsaufbau über vermittelnde Netzelemente (beispielsweise SIP Proxy Server, siehe Kapitel 3.3.3) funktioniert prinzipiell genauso wie bei der direkten Kommunikation (siehe Abb. 3.9). Der Nutzer (Bob) kontaktiert jetzt jedoch nicht direkt seinen gewünschten Gesprächspartner (Alice), sondern fragt die Verbindung bei einem Proxyserver seines Anbieters (atlanta.com) mit einer *INVITE*-Nachricht (Listing 3.1) an. Da der SIP Proxy Server die Nachricht jedoch entgegennimmt und weiterverarbeitet, erhält Bob vom ihm bereits eine vorläufige Statusnachricht *100 Trying*.

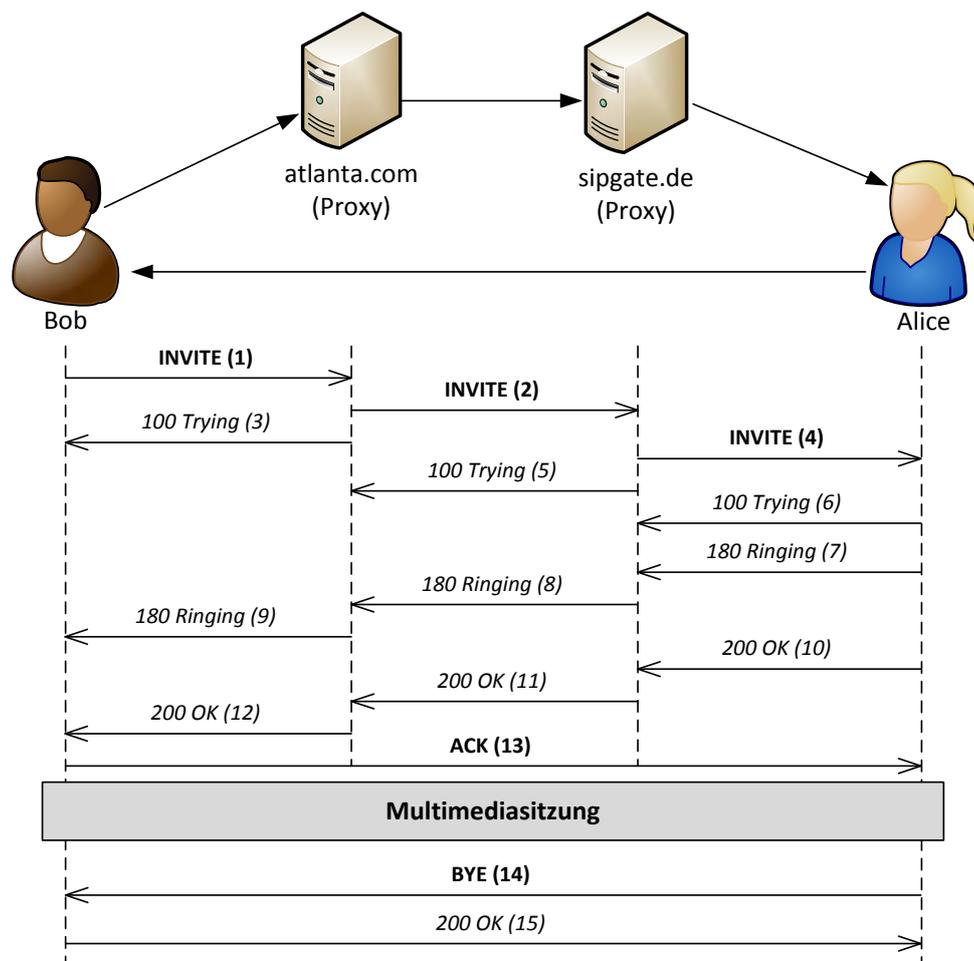


Abbildung 3.9: Aufbau einer SIP-Verbindung über SIP Proxy Server (auch als *SIP Trapez* bezeichnet)

Die INVITE-Nachricht wird von allen Proxys auf dem Weg zum gewünschten Gesprächspartner weitergeleitet. Die nächste vorläufige Statusnachricht *180 Ringing* erhält Bob, wenn die INVITE-Nachricht bei Alice eingegangen ist. Der zweite Schritt des Three-Way-Handshake, die finale Statusmitteilung *200 OK*, ist die letzte Nachricht, welche über den Proxy versendet wird. Die weiteren SIP-Nachrichten und der RTP-Datenstrom der Multimediasitzung nach der ACK-Anfrage werden, wenn möglich, zwischen den Nutzern direkt übertragen.

Listing 3.1: INVITE-Anfrage von Bob an Alice (vgl. Abb. 3.9)

```
INVITE sip:alice@sipgate.de SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bK776asdhds
Max-Forwards: 70
To: Alice <alice@sipgate.de>
From: Alice <sip:bob@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:bob@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142
(... SDP-Nachricht ...)
```

Die Medienparameter für die Einleitung der RTP-Verbindung werden gekapselt innerhalb der SIP-Nachricht als SDP-Paket versendet.

3.2.3 SDP - Session Description Protocol

Das Session Description Protocol (SDP) [RFC4566] wird zur Beschreibung von Medien innerhalb einer Kommunikationssitzung in paketbasierenden Netzwerken verwendet. SDP wurde nicht speziell für die Verwendung mit SIP entwickelt, ist aber gut für die Beschreibung der Mediensitzung, die eine SIP-Verbindung initiiert, geeignet.

Eine SDP-Nachricht enthält Informationen über den Anrufer (IP-Adresse oder Host), die verwendeten Medientypen und eine Aufzählung aller verfügbaren Codecs für die einzelnen Medientypen. Die einzelnen Parametertypen werden durch Kleinbuchstaben beschrieben.

```
v=0
o=Bob 1234 1234 IN IP4 pc33.atlanta.com
s=Video von 987654
c=IN IP4 host222.sipgate.de
```

```
t=0 0
m=audio 20000 RTP/AVP 8 97 4
a=rtpmap:8 PCMA/8000
a=rtpmap:97 iLBC/16000
a=rtpmap:4 G723/8000
m=video 20001 RTP/AVP 31
a=rtpmap:31 H261/90000
```

Diese Beispielnachricht enthält die Beschreibung einer Sitzung mit der ID 1234, die von Bob eingeleitet wurde (o-Feld, *Origin*). Die Sitzung hat den Namen „Video von 987653“ (s-Feld, *Session Name*) und ist an den Teilnehmer des Hosts `host222.sipgate.de` gerichtet (c-Feld, *Connection Information*). Als Medienströme werden Audio und Video (m-Feld, *media name*) angeboten, die auf dem angegebenen Port (für Audio Port 20000, Video 20001) empfangen werden. In dem Feld findet sich auch eine Auflistung der verfügbaren Codecs, die nach Bevorzugung sortiert sind. Die Codecs selbst werden jeweils über ein Feld mit der genannten Nummer und ihrer Bezeichnung beschrieben (a-Feld, *Attribute*). Dazu werden noch weitere Parameter, wie die Abtastfrequenz bei Audioverbindungen, angegeben.

3.2.4 Codecs

Unter einem Codec versteht man ein Paar aus einem Kodier- und einem Dekodieralgorithmus zur Umwandlung von Datenströmen. Im NGN können verschiedene Codecs für die Sprachkommunikation eingesetzt werden, welche sich durch die Sprachqualität, benötigte Bandbreite und die Verzögerung bei der Codierung unterscheiden (siehe Tabelle 3.1).

Der bekannteste Codec in der Telekommunikation ist der von der ITU-T spezifizierte *G.711* [TrWe07, S.45]. Dieser nutzt ein bandbegrenzt (300Hz bis $3,4\text{kHz}$), analoges Eingangssignal, das mit einer Abtastfrequenz von 8kHz und einer nichtlinearen Kennlinie (in Europa als a-Law bekannt) quantisiert und mit 8-bit-Worten digitalisiert wird. Der Codec bietet eine hohe Sprachqualität und eine sehr kurze Codierzeit, da jedes Sample einzeln kodiert wird. Dafür ist der Bandbreitenbedarf mit 64kbit/s relativ hoch. Viele Codecs verwenden Rahmen, die bis zu 240 Abtastwerte enthalten. Das bedeutet, dass bereits durch das Verpacken in einen Rahmen bei einer Abtastfrequenz von 8kHz eine Verzögerung (Blockdauer) von bis zu 30ms auftritt.

Codec	Algorithmus	Bitrate	Blockdauer	ges. Verzögerung	Sprachqualität
G.711	PCM ² , A-Law	64kBit/s	0,125ms	0,125ms	93
G.723.1	MP-MLQ ³	6,3kBit/s	30ms	37,5ms	78
G.723.1	ACELP ⁴	5,3 kBit/s	30ms	37,5ms	74
G.726	ADPCM ⁵	32kBit/s	0,125ms	0,125ms	86
iLBC	BILPC ⁶	13,3/15,2kBit/s	30/20ms	30/20ms	81/83
GSM-FR	RPE-LTP ⁶	13kBit/s	20ms	20ms	73
AMR	AMR ⁷	4,75-12,2kBit/s	20ms	20ms	max. 88

Tabelle 3.1: Sprachcodecs und zugehörige Kennwerte (Quelle: Auszug aus [TrWe07, S.46])

3.2.5 Real-Time Transport Protocol (RTP)

Zur Übertragung der codierten Nutzdaten wird das *Real-Time Transport Protocol* (RTP) genutzt, welches speziell für die Übertragung von Multimediadatenströmen wie Audio, Video oder Text über UDP/IP entwickelt wurde. Die Übertragung durch RTP ist dabei unabhängig vom Codec in welchem die Nutzdaten komprimiert sind. Die Steuerung und Kontrolle der Datenströme erfolgt über das *Real-Time Control Protocol* (RTCP), welches eine Kommunikation zwischen Quelle und Ziel zum Austausch von Quality-of-Service-Parameter ermöglicht, beispielsweise die Bandbreite an eine langsame Verbindung anzupassen.

Die einzelnen Rahmen der codierten Sprachdaten werden in RTP-Datenpaketen verpackt. Da diese Pakete je nach Codec mehrere Sprachrahmen aufnehmen können, wird mit dem Verschicken der Pakete so lang gewartet bis ein Paket gefüllt ist, was zu einer zusätzlichen Verzögerung beim Versenden führt. Die RTP-Datenpakete werden in Pakete des darunterliegenden Protokolls UDP verpackt und von diesem weiterbefördert. Diese Protokollverschachtelung benötigt ebenfalls eine Bearbeitungszeit und erhöht die Datenrate durch die Header, die jedes Protokoll den Paketen hinzufügt (Abb. 3.10). Durch die verbindungslosen Übertragung in einem IP-Netz kann die Anordnung der gesendeten Pakete auf dem Weg zum Empfänger durch die unterschiedliche Paketlaufzeit

²Pulse Code Modulation

³Multi-Pulse-Maximum-Likelihood Quantization

⁴Algebraic Code-Excited Linear Prediction

⁵Adaptive Differential Pulse Code Modulation

⁶Block-Independent Linear-Predictive Coding

⁷Regular Pulse Excitation Long Term Predictor

⁸Adaptive Multi Rate

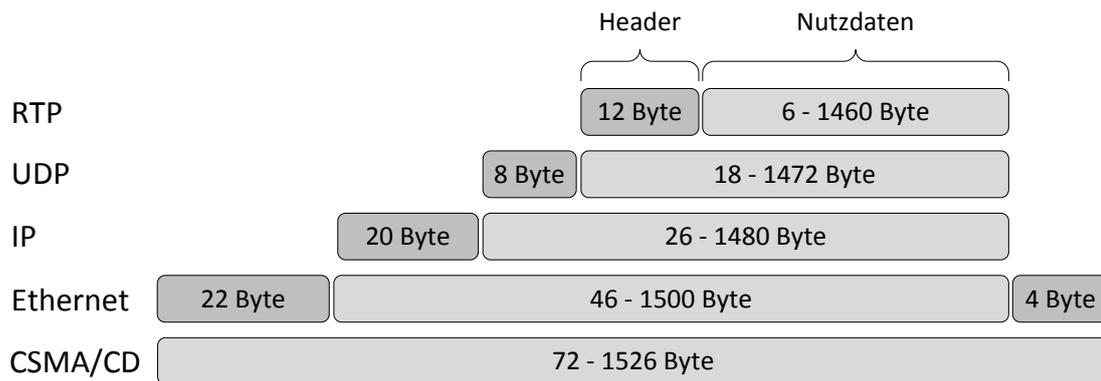


Abbildung 3.10: Protokollverschachtelung beim Versenden von RTP-Paketen (nach [TrWe07, S.48])

auf verschiedenen Übertragungswegen verändert werden. Beim Empfänger müssen die Pakete wieder in die entsprechende Reihenfolge gebracht werden, wofür bei der Übertragung via RTP entsprechende Informationen (Zeitstempel) zu den Sprachrahmen hinzugefügt werden. Dadurch kann der Empfänger die Sprachpakete wieder korrekt ordnen und ebenfalls erkennen, wenn Pakete verloren gegangen sind. Problematisch ist hier jedoch, wenn die Laufzeit der einzelnen Pakete sehr stark voneinander abweicht. An dieser Stelle wird der Decodieralgorithmus dazu gezwungen, die stark verzögerten Pakete zu verwerfen, um eine unterbrechungsfreie Wiedergabe mit akzeptabler Verzögerung zu ermöglichen. Die Gewährleistung einer hohen Dienstgüte ist ein zentrales Thema im NGN (siehe Kapitel 3.5.1).

3.3 Netzelemente

Die Netzelemente in einem NGN unterscheiden sich primär durch ihre logischen Funktionen und sind auch nur als logische Einheiten zu verstehen. In der Praxis sind in einem physikalischen System in der Regel mehrere logische Funktionen zusammengefasst.

3.3.1 User Agent

Der User Agent (UA) bildet die Schnittstelle zwischen einem Netzwerkdienst und dem Nutzer. Im NGN kann der User Agent ein Softphone sein, was als installierbares Programm auf einem PCs oder Smartphone läuft, oder eine Hardwarekomponente wie ein IP-Telefon (Abb. 3.3.1). Ein User Agent kann SIP-Verbindungen einleiten, terminieren

und bestimmte Medienströme (Audio, Video etc.) nutzen. Weitere Funktionen können der Versand von Kurznachrichten (Instant-Messaging) oder Präsenzdienste sein.



Abbildung 3.11: User Agent in Form eines SIP Softphones (links, X-Lite 3.0) und eines IP-Telefons (Cisco, Quelle⁹)

3.3.2 Registrar Server

Der Registrar Server ermöglicht die orts- und endgeräteunabhängige Kommunikation zwischen Netzteilnehmern. Wie in Abschnitt 3.2.2.1 beschrieben, ist erst durch die Nutzung von dauerhaften SIP URIs eine komfortable Kommunikation möglich. Die temporäre URI, welche vom Netzwerk und Endsystem des Nutzers abhängt, muss dem Anbieter der dauerhaften URI mitgeteilt werden. Dazu sendet der UA die SIP-Anfrage *REGISTER* mit der aktuellen Kontaktadresse (IP-Adresse oder Hostname) und der ständigen URI an den Registrar. Die temporäre URI wird zusammen mit der dauerhaften URI an den Location Server übergeben. Die Anfrage wird anschließend durch die Statusmitteilung *200 OK* quittiert. Im Normalfall entspricht die Registrierung beim Anbieter der Aktivierung des Nutzerkontos, so dass anschließend andere Netzteilnehmer den Nutzer unter seiner ständigen URI erreichen können.

3.3.3 Proxy Server

Der Proxy Server ist für die Weiterleitung von SIP-Nachrichten zuständig. Die Routinentscheidungen für die SIP-Pakete trifft der Proxy anhand der angefragten URI oder

⁹Cisco Presseinformationen Bidlarchiv http://www.cisco.com/web/DE/presse/bildarchiv_home.html

der Kontaktadresse in Statusmitteilungen. Eine wichtige Aufgabe liegt für den Proxy im Routing von Paketen an eine dauerhafte URI, welche an eine temporäre URI eines Nutzers weitergeleitet werden müssen. Dazu nutzt der Proxy den Location Server, welcher die Zusammenhänge von temporären und dauerhaften URIs herstellt.

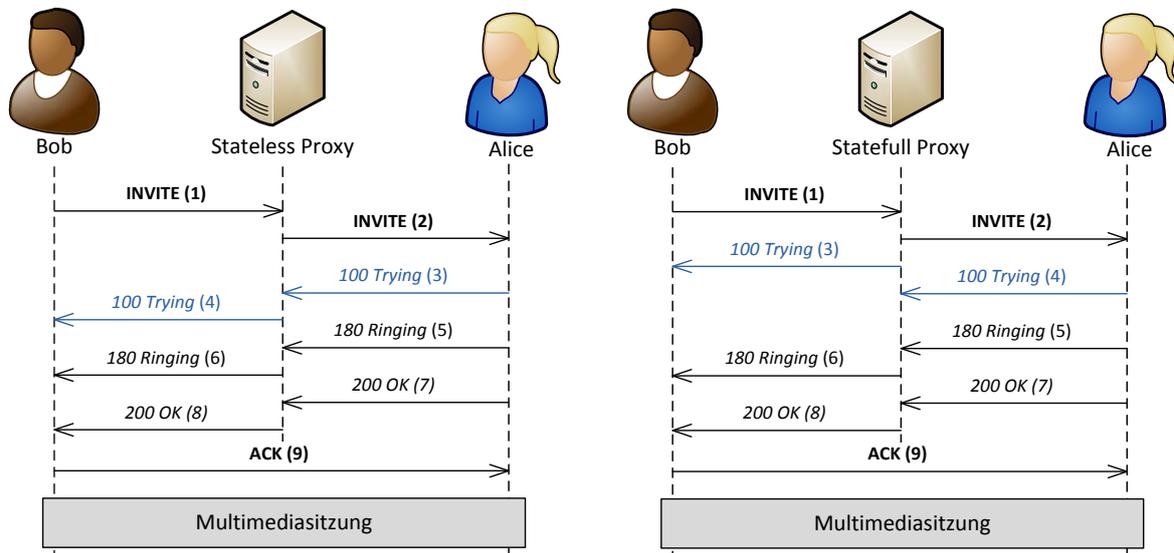


Abbildung 3.12: SIP-Sitzungsaufbau über einen *Stateless* (links) und einen *Statefull*-Proxy-Server (nach [TrWe07, S.184 f.])

Bei der Weiterleitung von SIP-Paketen wird zwischen *Statefull* und *Stateless* Proxy Servern unterschieden. Das bereits besprochene Beispiel in Abschnitt 3.2.2.3 (Abb. 3.9, SIP-Trapez) stellt einen *Stateless*-Proxy dar. Dieser leitet die SIP-Pakete einfach anhand seiner Routingentscheidungen weiter und greift nicht weiter in die Kommunikation ein. Er kann keine Statusmitteilungen versenden und speichert keine SIP-Pakete zwischen. Der *Statefull*-Proxy hingegen greift direkt in die Kommunikation zwischen zwei UAs ein (Abb. 3.12). Er speichert den Transaktionsstatus aller SIP-Anfragen und erzeugt selbstständig vorübergehende Statusmitteilungen (z.B. *100 Trying*) oder SIP-Anfragen wie *CANCEL*.

3.3.4 Location Server (HSS)

Der Location Server, in einigen Netzmodellen auch als Home Subscriber Server (HSS) bezeichnet, speichert die Beziehungen zwischen temporären und dauerhaften URIs in einer Datenbank. Diese Datensätze erhält er vom Registrar Server, wenn sich ein UA dort registriert. Der Location Server ist kein direktes Netzelement im Sinne einer SIP-Sitzung, da er keine SIP-Nachrichten verarbeitet. Die Kommunikation mit dem Regis-

tration Server und den Proxy Servern findet über andere Protokolle, wie Diameter oder LDAP¹⁰, statt.

3.3.5 Gateways

Gateways dienen zur Anbindung von anderen Netzen an das NGN, wenn dort andere Signalisierungsprotokolle und Nutzdatenprotokolle oder -codecs verwendet werden. Sie stellen ein sehr wichtiges Netzelement dar, da durch sie die bestehenden Kommunikationsnetze wie ISDN oder GSM in das NGN eingebunden werden können. Sie ermöglichen eine bidirektionale Kommunikation zwischen den NGN-Nutzern und den Nutzern von externen Kommunikationsnetzen.

Die Gateways bestehen aus zwei logischen Einheiten, einem *Signalling Gateway* (SGW) und einem *Media Gateway* (MGW). Das SGW konvertiert die Signalisierung im NGN (SIP) in das Format der Signalisierung des externen Netzes (z.B. SS7 bei GSM). Das MGW konvertiert den Nutzdatenstrom der Mediensitzung in das Format des externen Netzes. Aus Kostengründen werden beide Elemente häufig zusammengefasst.

Bei der Verbindung von zwei IP-basierenden Netzen, die auf SIP und H.323 basieren, müssen nur die Signalisierungsabläufe konvertiert werden. Die Mediensitzung basiert bei beiden Protokollfamilien auf RTP und braucht nicht verändert werden, wenn ein auf beiden Seiten unterstützter Codec verwendet wird.

Je nach Funktion und Einsatzort wird zwischen verschiedenen Gatewaytypen unterschieden. Ein Trunking Gateway bindet ein anderes Kernnetz, wie GSM oder ISDN, an. Das Access Gateway verbindet ein Zugangnetz, z.B. DSL oder WiMAX, mit dem NGN. Über ein Residential Gateway können lokale Netze oder Endgeräte an das NGN angeschlossen werden.

3.3.6 Back-to-Back User Agent

Ein *Back-to-Back User Agent* (B2BUA) ist ein Element im NGN, der zur direkten SIP-Kommunikation zwischen zwei Teilnehmern nicht benötigt wird. Der B2BUA ermöglicht jedoch die direkte Steuerung eines Gesprächs über die Funktionalitäten eines Proxy Servers hinaus, denn er hat die Möglichkeit wie ein normaler UA SIP-Anfragen wie *INVITE* oder *BYE* zu generieren [Zmor09]. Damit kann er ein Gespräch zwischen zwei Teilnehmern einzuleiten und aktiv steuern.

¹⁰Lightweight Directory Access Protocol (LDAP) für Abfragen und Modifikationen einer verteilten Datenbank, festgelegt in [RFC2251].

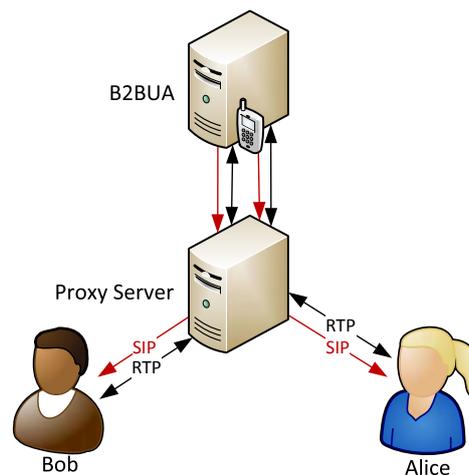


Abbildung 3.13: Ein *Back-to-Back User Agent* kann im Gegensatz zu einem Proxy eine Verbindung zwischen zwei Teilnehmern aktiv steuern oder sogar einleiten. Bei Bedarf kann er auch die Medienströme (RTP) terminieren und so beispielsweise NAT an Netzwerkgrenzen überwinden.

Er tritt gegenüber beiden Teilnehmern jeweils als UA auf und empfängt ihre Anfragen und steuert die Mediensitzung. Die Mediensitzung selbst kann jedoch auch direkt zwischen den Teilnehmern aufgebaut werden (Abb. 3.13). Eine mögliche Anwendung dafür ist *Click-to-Dial* (siehe Kapitel 4.2.3), was als Third-Party ein Gespräch mit einem Teilnehmer einleitet (etwa durch einen Klick auf einen Kontakt in einem Mailprogramm) und den Nutzer anschließend mit dem gewünschten Gesprächspartner verbindet. Ein B2BUA kann aber auch zur Verbindung von zwei Netzwerken genutzt werden. Das wird verwendet um die Netztopologie vor den Nutzern des anderen Netzes zu verbergen, oder zur Überwindung der Network Address Translation an den Grenzen von Netzwerken, was für SIP-Verbindungen ein generelles Problem ist. Dabei muss natürlich auch der RTP-Datenstrom vom B2BUA terminiert werden, da diese Grenzen für diesen genauso gelten wie für die SIP-Signalisierung.

3.3.7 Konferenzserver

Die Kommunikation über die SIP-Protokollfamilie beinhaltet auch eine Möglichkeit für Konferenzgespräche. Da eine SIP-Sitzung jedoch nur zwischen zwei Teilnehmern etabliert werden kann, muss hierfür auf einen Konferenzserver (*Multipoint Control Unit*) zurückgegriffen werden [RFC4353].

Der Konferenzserver besteht aus zwei Instanzen, welche die unterschiedlichen Medienströme terminieren (Abb. 3.14). Der *Focus* tritt als User Agent auf und steuert

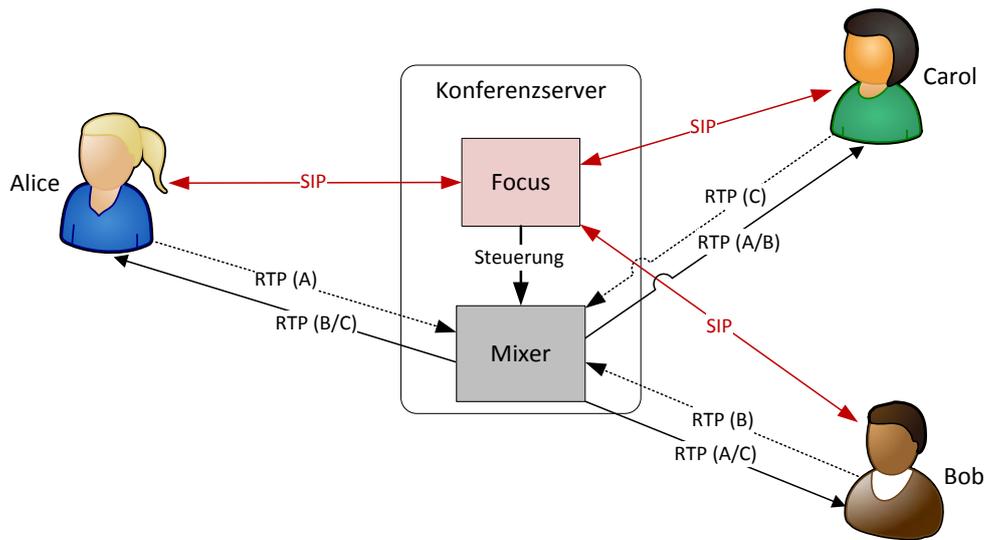


Abbildung 3.14: Ein Konferenzgespräch braucht bei SIP einen Konferenzserver, da eine SIP-Sitzung nur zwischen zwei Endsystemen bestehen kann (frei nach [TrWe07, S.226]).

die SIP-Sitzungen mit den Teilnehmern der Konferenz. Die Mediensitzungen der Teilnehmer werden durch die *Mixer*-Instanz terminiert, welche direkt vom Focus gesteuert wird. Der Mixer empfängt die einzelnen Datenströme der Nutzer und multiplext sie pro Nutzer zu einem neuen Datenstrom. Theoretisch ist der Anzahl der Konferenzteilnehmer durch Focus und Mixer keine Grenze gesetzt, wobei praktisch die Bandbreite und die Rechenkapazität des Konferenzservers die Anzahl der gleichzeitigen Konferenzen und der Teilnehmer limitiert. Typischerweise können die Mediendatenströme alle im NGN unterstützten Medien beinhalten (Audio, Video, Text etc.).

3.3.8 Presence Server

Die Präsenzfunktion dient dazu, bereits vor Verbindungsaufbau zu erkennen, ob der gewünschte Gesprächspartner erreichbar ist. Dieses Komfortmerkmal wird heute bereits in den meisten Instant-Messaging-Protokollen (IM) eingesetzt.

Im NGN wird diese Funktionalität über einen *Presence Server* bereitgestellt, bei welchem der Nutzer über eine NOTIFY-Nachricht den Status eines anderen Teilnehmers abonnieren kann. Ändert sich der Erreichbarkeitsstatus des anderen Teilnehmers, erhält der Nutzer eine Benachrichtigung. Für SIP wurden in [RFC3265] eine Reihe von Ereignismitteilungen definiert.

Dabei wird zwischen Basis-Nachrichten, die nur den Status `open` oder `close` besitzen, und erweiterten Nachrichten, die genauere Informationen über den Online-Status

enthalten, unterschieden. Die Mitteilung über den Status erfolgt innerhalb der SIP-Nachricht über das *Presence Information Data Format* (PDIF) [RFC3863] in Form von XML.

```
<?xml version="1.0" encoding="UTF-8" (...) ?>
(...)
  <tuple id="user">
    <status>
      <basic>open</basic>
    </status>
  </tuple>
```

Der Presence Server speichert den Zustand aller angemeldeten Nutzer, die diesen Dienst nutzen und verteilt entsprechend der Abonnements die Nachrichten im Netz.

3.3.9 Anwendungsserver

Die Grenze von den einzelnen Netzelementen, wie dem B2BUA oder dem Konferenzserver, zum Anwendungsserver sind fließend, da ein Anwendungsserver (AS) viele dieser Funktionalitäten ebenfalls implementieren kann. Der Anwendungsserver ist das Netzelement, welches das NGN um alle Funktionen und Mehrwertdienste erweitern soll, die in den grundlegenden Spezifikationen für die sitzungsorientierte Kommunikation nicht enthalten sind. Im NGN können alle Elemente der Anwendungsschicht (siehe Abb. 3.3) unter dem Oberbegriff Anwendungsserver zusammengefasst werden. Häufig werden auch *Media Server* und *Presence Server* zur Anwendungsschicht gezählt.

Der Anwendungsserver stellt eine *Service Delivery Platform* (SDP, siehe Kapitel 5.1.1) dar, welche eine einheitliche Programmierschnittstelle (API) für Anwendungen bereitstellt. An das NGN ist er über einen eigenen UA angebunden, über welchen die Dienste mit den Netzteilnehmern interagieren können oder die Netzteilnehmer die Dienste aufrufen können. Da Anwendungsserver für die Erzeugung von Mehrwertdiensten im NGN von zentraler Bedeutung sind, werden Funktionen und Konzepte in Kapitel 5 näher untersucht.

Weil durch den großen Preisverfall bei der herkömmlichen Telefonie die Erträge der Netzbetreiber stark zurückgegangen sind, werden für sie im NGN Mehrwertdienste und neue Anwendungen von großem Interesse sein, da Zukunft vermutlich ein Großteil der Einnahmen der Anbieter daraus generiert wird [Poll09] [TrWe07, S.229].

3.3.10 weitere Netzelemente

Es existieren noch einige weitere Netzelemente im NGN, die bis hier nicht näher erläutert wurden, da sie teilweise in anderen funktionalen Einheiten bereits enthalten sind.

Der **Media Server** ist ein spezieller Anwendungsserver, der für die Auslieferung von Mediendatenströmen (Streaming-Server) oder für die Konvertierung von Datenströmen zuständig ist. Prinzipiell kann er als Media-Controller auch die Funktion der Medienkonvertierung zwischen zwei paketbasierenden Netzwerken eingesetzt werden und ist dann ein Teil des Gateways. Häufig kommuniziert ein Mediaserver nur über RTP, während die zugehörige SIP-Verbindung über einen anderen Anwendungsserver läuft, von dem der Mediaserver gesteuert wird.

Ein **Redirect Server** wird für die Beantwortung von Anfragen mit dem Statuscode 3xx (Weiterleitung) eingesetzt. Mit ihm können beispielsweise Rufumleitungen von einer SIP-URI zu einer anderen realisiert werden.

Ein **Session Border Controller** verbindet zwei SIP-Kernnetze zweier unterschiedlicher Provider miteinander. Er unterbricht teilweise die Ende-zu-Ende-Transparenz die in SIP definiert ist und bietet die Möglichkeit den Verkehr zwischen zwei Netzen nach verschiedenen Vorgaben (Dienstgütevereinbarungen, gesetzlichen Vorgaben etc.) zu kontrollieren, zu steuern und abzurechnen.

3.4 Zugangsnetze

Das Ziel der NGN-Spezifikationen ist, dass alle bisherigen Netze als Zugangnetz direkt oder über ein Gateway angeschlossen werden können. Die erste IMS-Spezifikation *UMTS Release 5* sah von Seiten der 3GPP nur sogenannte *Radio Access Networks* (RAN) als Zugangnetze vor. Da in diesem Konzept der Nutzdatentransport bereits von den Kontrollfunktionen und der Anwendungsschicht getrennt waren, konnte die ETSI dieses Konzept auch für die Anbindung anderer Netze verwenden. So wurde in Release 6 andere drahtlose Netzwerke wie WiMAX und WLAN eingeführt, und in Release 7 kabelgebundene Zugangnetze integriert. Prinzipiell sind alle paketbasierenden Breitbandnetze (siehe Kapitel 2.3) und Mobilfunknetze als Zugangnetze nutzbar. Über Gateways ist das NGN auch über analoge Telefone (POTS) oder das digitale Telefonnetz (ISDN) nutzbar.

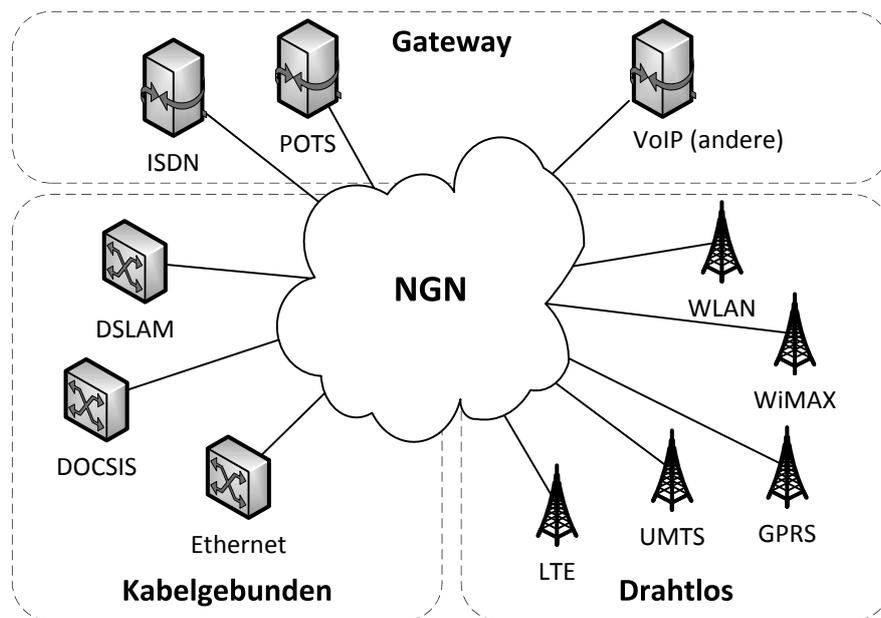


Abbildung 3.15: Das NGN kann über verschiedene drahtlose und kabelgebundene Zugangsetze genutzt werden. Der Zugang anderer Telekommunikationsnetze ist über Gateways möglich.

3.5 Netzmerkmale

3.5.1 Dienstgütegarantien (QoS)

Die Dienstgüte (QoS von engl. *Quality-of-Service*) ist in paketvermittelnden Telekommunikationsnetzen ein wichtiges Leistungsmerkmal. Ein einfaches paketbasierendes Netz bietet gar keine Dienstgütegarantien, sondern leitet Daten einfach schnellstmöglich und ohne Fehler nach dem Best-Effort-Prinzip weiter. Geringe Verzögerungen und konstante Laufzeiten, wie sie für Echtzeitkommunikation benötigt werden, können damit nicht gewährleistet werden [TrWe07, S.108]. Die maximale Verzögerung, die für den Nutzer akzeptabel ist, wird von der ITU-T mit 150 – 400ms angegeben [G114].

Um QoS in Paketnetzen trotzdem bereitstellen zu können, existieren verschiedene Möglichkeiten, wie die Überdimensionierung des Netzes oder die Nutzung von ATM, Integrated Services (IntServ), Differentiated Services (DiffServ) oder Multiprotocol Label Switching (MPLS).

Bisher wird häufig die Überdimensionierung des Netzes als einfachste Möglichkeit genutzt, die Netze den Anforderungen anzupassen. Diese Variante benötigt keine Anpassung der Netzstruktur und beschleunigt alle genutzten Dienste im Netz gleichermaßen.

3.5.1.1 ATM

Asynchronous Transfer Mode (ATM) ist eine Übertragungstechnik, die leitungsvermittelnde und paketbasierende Datenübertragung unterstützt. ATM basiert auf sogenannten Zellen, Paketen mit einer festen Länge, die zu Zellströmen (Zellen mit dem gleichen Ziel) zusammengefasst werden. Zellströme werden wiederum zu Virtual Paths (VP) und Virtual Channels (VC) zusammengefasst, solange sie das gleiche Ziel im Netz haben. Die Vermittlung im Netz leitet nun alle Zellen, die zum gleichen VP und VC gehören, über den gleichen Übertragungsweg. Damit erhält man eine durchgehende Verbindung, da alle folgenden Datenpakete die gleiche Verbindung nutzen. ATM garantiert außerdem über Traffic Contracts bestimmte Bitraten für einzelne Zellströme.

3.5.1.2 MPLS

Multiprotocol Label Switching (MPLS) ist eine Technik zum vereinfachten Routing von Datenpaketen, die eine verbindungsorientierte Übermittlung von Datenpaketen ermöglicht. Dabei wird das Paket in den Routern nicht mehr anhand des IP-Headers, sondern auf Basis seines Labels weitergeleitet. Dazu muss vorher mit dem Resource Reservation Protocol (RSVP) ein entsprechender Pfad gesucht und in den Routern bekannt gemacht werden. Damit werden alle Datenpakete über den gleichen Übertragungsweg verschickt.

3.5.1.3 IntServ und DiffServ

IntServ und DiffServ sind QoS-Verfahren zur Priorisierung von IP-Paketen. Für IntServ wird vor dem Verbindungsaufbau über das RSVP eine Anfrage auf eine Bandbreitenreservierung auf einem Pfad zum Ziel durchgeführt. Dazu muss jeder Router seine verfügbare Bandbreite genau protokollieren und alle Netzelemente IntServ unterstützen. Die zugesicherten Dienstgütekriterien gelten für die gesamte Sitzungsdauer. DiffServ dagegen arbeitet nicht mit garantierten Bandbreiten, sondern mit Paketprioritäten. Dazu muss der Sender alle Pakete einer Verbindung mit einer Prioritätsklasse kennzeichnen, und die Router auf dem Weg entscheiden innerhalb der Warteschlangen anhand dieser Klassen über die Bearbeitungsgeschwindigkeit. Problematisch dabei ist, dass die Korrektheit der Kennzeichnung kaum geprüft werden kann.

Die Garantie von Dienstgütekriterien ist in IP-basierenden Netzen nur bedingt möglich. Am häufigsten werden dafür DiffServ in Verbindung mit MPLS eingesetzt [TrWe07, S.121]. Generell ist QoS aber eines der wichtigsten Probleme das im Zuge der Einfüh-

nung des NGN gelöst werden muss, um eine Akzeptanz bei Nutzern zu erreichen, welche die Dienstgüte eines leitungsvermittelnden Netzes erwarten.

3.5.2 Mobilität

Die Mobilität nimmt in allen NGN-Konzepten eine zentrale Rolle ein. Besonders durch die Verbindung von mehreren Zugangsnetzen werden hohe Anforderungen an die Mobilität gestellt. Insgesamt kann man vier Arten der Mobilität in NGNs unterscheiden.

3.5.2.1 Persönliche Mobilität

Die persönliche Mobilität bezeichnet die Erreichbarkeit eines Nutzers über eine ständige URI (siehe Kapitel 3.2.2.1). Diese Erreichbarkeit soll unabhängig von seinem Aufenthaltsort, seinem Endgerät und seinem Zugangnetz gewährleistet sein. Diese Funktionalität wird durch die Registrierung des genutzten UAs beim Registrar Server (siehe Kapitel 3.3.2) gewährleistet. Eingehende Verbindungen werden anschließend von einem Proxy Server auf die temporäre URI des Nutzers weitergeleitet.

3.5.2.2 Endgerätemobilität

Unter Endgerätemobilität versteht man die Unabhängigkeit eines Endgerätes vom gerade verwendeten Zugangnetz. Dies ist besonders im Bereich von mobilen Endgeräten, die zwischen verschiedenen Funknetzen wechseln können, von Interesse. Die Realisierung der Endgerätemobilität ist besonders für bestehende Sitzungen sehr komplex. Da sich mit dem Wechsel zwischen zwei Netzen häufig auch die IP-Adresse des Endgerätes ändern kann, wird dafür eine Implementierung von *Mobile IP* [RFC3344] benötigt, um einen nahtlosen Transfer der Sitzung (Handover) zu ermöglichen. Für die Endgerätemobilität müssen aber auch die Anwendungsfälle für die Gesprächsbereitschaft oder des kurzzeitigen Ausfalls der Funkverbindung gelöst werden. Durch die Endgerätemobilität können aber auch völlig neue Szenarien realisiert werden. Beispielsweise kann in Gebieten mit schlechter Funkversorgung (ländliche Gebiete, Stahlbetonbauten etc.) oder in überlasteten Funkzellen durch den Übergang vom öffentlichen UMTS-Netz in private Femto-Zellen die Übertragungsqualität deutlich gesteigert werden [Ziva09]. Dazu wird ein UMTS-Router, der eine eigene kleine UMTS-Funkzelle bereitstellt, mit einem kabelgebundenen Breitbandanschluss verbunden, über welchen die Verbindungen dann in das Netz des jeweiligen Anbieters übertragen werden.

3.5.2.3 Sitzungsmobilität

Die Sitzungsmobilität erlaubt innerhalb einer Kommunikationssitzung das Endgerät, und damit seine temporäre URI, zu wechseln. Um eine Sitzung zu einem anderen Endgerät zu übertragen, kann die Sitzung entweder vom gerade aktiven UA einfach weitergeleitet werden, oder sie wird über die SIP-Nachricht REFER vom aktuellen UA zum nächsten übergeben. Ein gängiges Szenario dafür ist der Wechsel von einem mobilen zu einem stationären Endgerät, welches dann beispielsweise Videoübertragung als zusätzliches Medium innerhalb der Sitzung unterstützt.

3.5.2.4 Dienstmobilität

Über die Dienstmobilität wird sichergestellt, dass der Nutzer unabhängig von Zugangsnetz und Standort jederzeit alle Dienste im NGN nutzen kann. Der Hintergrund dafür ist, dass das NGN ein offenes Netz sein soll, vergleichbar mit dem Internet, in dem jeder Diensteanbieter seine Dienste uneingeschränkt für alle Netzteilnehmer bereitstellen kann.

3.5.3 Sicherheit

Das NGN besitzt als IP-basierendes Netz generell alle Schwachstellen, die heute bei IP-Netzen auch schon auftreten. Attacken wie IP Spoofing (gefälschte Sendeadressen), Flooding (Überfluten von Endsystemen) oder Hijacking (Übernahme von Netzelementen oder Endgeräten) sind Gefahren die jedes System im Netz betreffen [Detk07, S.101].

Speziell in einem Kommunikationsnetz sind dabei auch eine Reihe weiterer Angriffe denkbar. Spam-over-Internet-Telephony (SPIT) wurde in diesem Zusammenhang schon häufiger erwähnt und wird bei einem offenen Netz wie den Mailservern unweigerlich irgendwann ein Problem werden. Aus diesem Grund soll eine Kommunikation im NGN nur nach einer Authentifizierung bei einem Service-Provider (siehe Kapitel 4.2.1) möglich sein. Damit wird eine Nachverfolgbarkeit, ähnlich wie im heutigen Telefonnetz, möglich. Das Ausspähen von Nutzerdaten kann über Angriffe wie „Man in the Middle“, Sniffing oder Phreaking geschehen [Detk07, S.104 ff.], was ebenfalls ein durch ein Authentifizierungssystem eingeschränkt werden kann.

Dazu können Schwachstellen von Seiten des Providers (z.B. Mailboxfehler [Detk07, S.102]) oder Fehler in der Software (Sicherheitslücken im UA) kommen, welche direkt das eigene Endgerät betreffen. Viren für mobile Endgeräte mit verbreiteten Betriebssystemen existieren bereits [Bach09].

3.5.4 Dienste

Die Erzeugung von Diensten im NGN wird nicht wie im Intelligenten Netz (siehe Kapitel 2.2) durch die starren Strukturen der Systemkomponenten begrenzt. Als IP-basierendes Netz wird die Entwicklung und Bereitstellung von Diensten ähnlich schnell und kostengünstig möglich sein, wie im Internet. Dies wird auch Auswirkungen auf die wirtschaftliche Nutzung dieser Dienste haben.

*Nischenanwendungen und kurzlebige Anwendungen wurden von den Netzbetreiber häufig übersehen und als kommerziell unbrauchbar betrachtet. Als Netzbetreiber lässt sich der Aufwand zur Bereitstellung von Diensten, für eine handvoll Nutzer, die nur für wenige Tage genutzt werden, nicht rechtfertigen. Das Potential für Innovationen außerhalb des Bereiches der Netzbetreiber ist noch vollständig unerschlossen.*¹¹

Das bedeutet, dass durch die wegfallende Begrenzung zu den Netzbetreibern als alleinige Dienstleister im Telekommunikationsnetz eine neue Art von Diensten möglich ist. Diese können nicht mehr nur von Wirtschaftsunternehmen sondern auch durch gemeinnützige Projekte, wie beispielsweise Wikimedia (der Trägerverein des freien Online-Lexikons Wikipedia) oder durch kleine Startupunternehmen eingeführt werden. Die traditionellen Geschäftsmodelle der Festnetzbetreiber geraten damit stark unter Druck, so dass diese allein mit der Kostensenkung durch die Einführung eines IP-basierenden Telekommunikationsnetzes, ohne neue attraktive Dienste, die auf das NGN zugeschnitten sind, dem Wettbewerb nicht standhalten werden [Poll09].

Technisch lassen sich die Anforderungen an zukünftige Dienste im NGN folgendermaßen zusammenfassen [LeTO07a] :

- einfache, schnelle und kostengünstige Entwicklung und Bereitstellung
- neue innovative Dienste die auf das NGN zugeschnitten sind
- Verknüpfung unterschiedlicher Medien (Multimedia)
- Dienstmobilität
- Sicherheit

3.6 Zusammenfassung

Das NGN ist ein paketbasierendes Telekommunikationsnetz, welches durch die Trennung von Transport, Verbindungssteuerung und Anwendungen eine flexible Basis für

¹¹Eigenständige Übersetzung nach [BBDU00]

die Telekommunikation darstellt.

Das größte Problem in diesem Paketnetz ist die Bereitstellung von QoS-Garantien. Zum einen kollidiert dieser Grundsatz mit der Netzneutralität, welche eine Gleichberechtigung aller Datenströme erfordert. Zum anderen müssen alle Netzelemente die Techniken zur Bereitstellung von QoS in IP-Netzen beherrschen. Es existieren zwar bereits andere Ansätze zur Verbesserung der Übertragung von Echtzeitdatenströmen ohne Einführung von QoS-Protokollen [Robe09], welche aber aufgrund der geringen Flexibilität der Netzbetreiber und dem derzeit noch kostengünstigen Ausbau der Bandbreite noch keine Anwendung finden. Die ständig steigende verfügbare Bandbreite, auch im Bereich der Zugangsnetze, wird diesen Ausbau auch in nächster Zeit weiter begünstigen. Es ist zu erwarten, dass innerhalb eines Jahres bereits Netzzugänge für Endnutzer mit Bandbreiten von 100MBit/s verfügbar sein werden (Abb. 3.16).

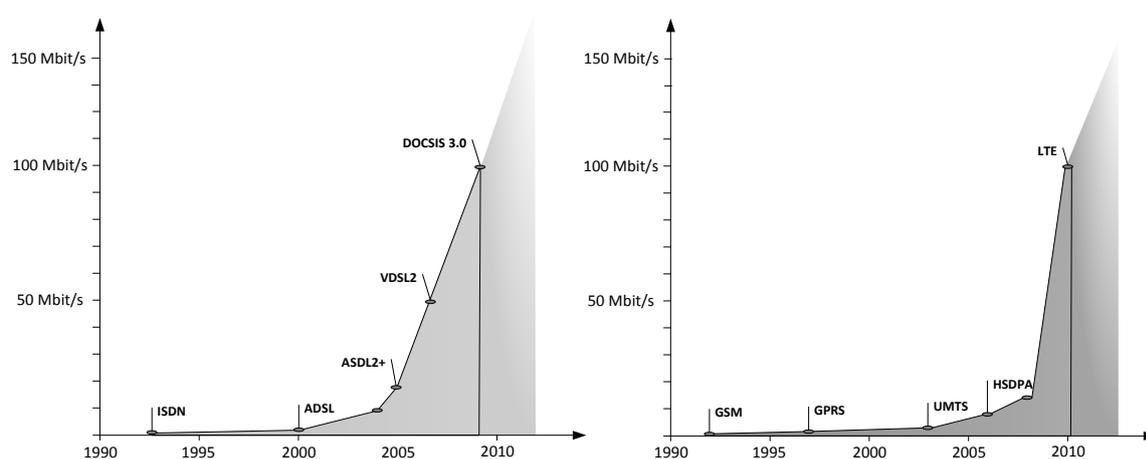


Abbildung 3.16: Die Entwicklung der Bandbreite in den kabelgebundenen (links) und kabellosen Netzen wird voraussichtlich im Jahr 2010 die Schwelle von 100MBit/s überschreiten.)

Im Bereich der Sprach- und Breitbanddienste herrscht im NGN für die Festnetzbetreiber eine große Konkurrenz, da neue Mitbewerber wie Kabelfernsehanbieter, Satelliten und Mobilfunkbetreiber durch eigene Zugangsnetze und die generelle Mobilität der Dienste plötzlich die gleichen Dienstleistungen anbieten können.

Somit ist der Ausblick auf zukünftige Geschäftsmodelle gleichzeitig einer der interessantesten Aspekte am gesamten NGN-Konzept - die Einführung neuer und innovativer Dienste, welche durch die neuen technischen Möglichkeiten ein wesentlich breiteres Spektrum abdecken werden, als die Dienste der letzten Generation der Telekommunikationsnetze.

4 Dienste im NGN

4.1 Definition (Dienst)

Allgemeinsprachlich ist ein Dienst eine „*eintönige, langweilige Erfüllung von [beruflichen] Pflichten*“ [Dude06, S.400]. In der Informationstechnik bedeutet das, dass ein Dienst eine wiederkehrende Dienstleistung ist, die von einem System für andere Systeme bereitstellt. Je nach Art des Systems gibt es unterschiedliche Definitionen für einen Dienst, welcher entsprechend der Sichtweise in Telekommunikationsnetzen in verschiedenen Schichten des OSI-Referenzmodells (Abb. 4.1) erbracht wird.

Kommunikationsdienst Im Bereich der Kommunikationsnetze kann man einen Dienst als Fähigkeit des Netzes betrachten, eine bestimmte Art von Informationen zu übertragen. Lange Zeit wurden die Netze auf jeweils nur einen Kommunikationsdienst konzipiert, wie das Telefonnetz, was nur die Übertragung von Sprachkommunikation auslegt ist. Mit ISDN wurde erstmals der Ansatz *ein Netz - mehrere Dienste* umgesetzt. Die meisten Kommunikationsdienste wie ATM, ISDN¹ oder X.212 erbringen Dienste der Vermittlungsschicht (OSI-Modell 3).

Netzwerkdienst In einem Datennetzwerk ist ein Dienst eine Funktion, welche der Nutzer als Anwendung nutzt. D.h. er greift mit einem Client auf den Dienst, der von einem Dienstanbieter (hier Server) bereitgestellt wird, zu. Solche Dienst können einfache Datei- (FTP), Druck-, Datenbank- oder HTTP-Dienste (Internet) sein. Diese Netzwerkdienste nutzen Protokolle der Transportschicht (OSI-Modell 4) und erzeugen selbst die Funktionen der Sitzungsschicht. Die Protokolle SIP und RTP, welche die Basis aller Dienstleistungen im NGN sind, gehören damit ebenfalls zur Sitzungsschicht.

Webservice Im Internet versteht man unter einem Dienst beispielsweise einen Webservice. Das sind Anwendungen, welche über eine URI aufgerufen werden können und

¹Die Vermittlung der Verbindung erfolgt bei ISDN über eine separate Leitung (D-Kanal), die bereitgestellte Verbindung selbst entspricht im ISDN-Modell eigentlich der OSI-Schicht 4 [Seit07].

die über XML-Nachrichten mit anderen Anwendungen kommunizieren können. Solche Dienste können Suchmaschinen (API für Google oder Wolfram Alpha), CRM-Systeme (SAP NetWeaver) oder Buchungssysteme (Fluggesellschaften oder Veranstaltungsservice) sein, wobei die Daten dann in der Zielanwendung dargestellt werden. Diese Dienste nutzen bereits implementierte Elemente der Anwendungsschicht, sind aber als Schnittstelle in der Darstellungsschicht definiert (z.B. WSDL, SOAP).

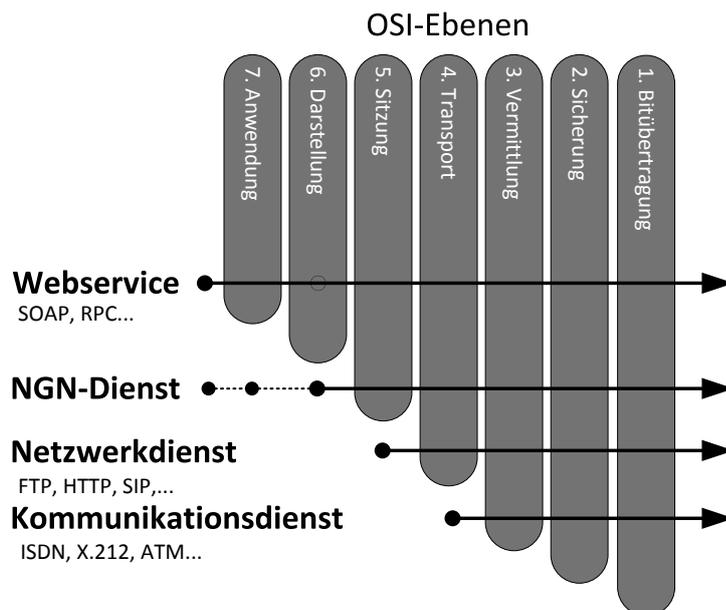


Abbildung 4.1: Ebenen im OSI-Modell, auf denen Dienste in Kommunikationsnetzen bereitgestellt werden. Dienste im NGN können je nach Funktion in verschiedenen Ebenen des OSI-Referenzmodells eingeordnet werden.

Dienste im NGN Da alle Dienste im NGN als Basis die Protokolle SIP und RTP nutzen müssen, damit eine Kommunikation mit den Netzelementen und Endsystemen möglich ist, bauen sie zwangsläufig auf Elemente der Vermittlungsschicht auf. Sie können aber, ähnlich wie Webservices, auch in höheren Schichten angesiedelt sein.

4.2 Klassifizierungsschemata für Dienste

Für Dienste in Next Generation Networks wird, durch die veränderte Nutzung des Netzes und das wesentlich breitere Angebot an Diensten, eine neue Klassifizierung benötigt. Die herkömmliche Unterscheidung aus dem Telefonnetz nach Basisdienstleistungen (Telefon- und Telefaxverbindungen) und Mehrwertdiensten (Servicrufnummern,

universelle Rufnummern) trifft nur noch bedingt zu, da die einfache Sprachverbindung zwischen zwei Teilnehmern nicht mehr die dominierende Dienstleistung ist, sondern ein Dienst unter vielen. Als Ausgangspunkt für eine tiefergehende Kategorisierung kann hier die Einteilung der Dienste im intelligenten Netz (siehe Kapitel 2.2) herangezogen werden.

Für die Einteilung der Dienste im NGN werden in der Literatur verschiedenen Kriterien herangezogen (Abb. 4.2), welche je nach Standpunkt des Betrachters verschiedene Ansätze verfolgen. Hier sollen einige ausgewählte Klassifizierungsschemata vorgestellt werden, um die untersuchten Dienste im NGN anschließend entsprechend ordnen zu können.

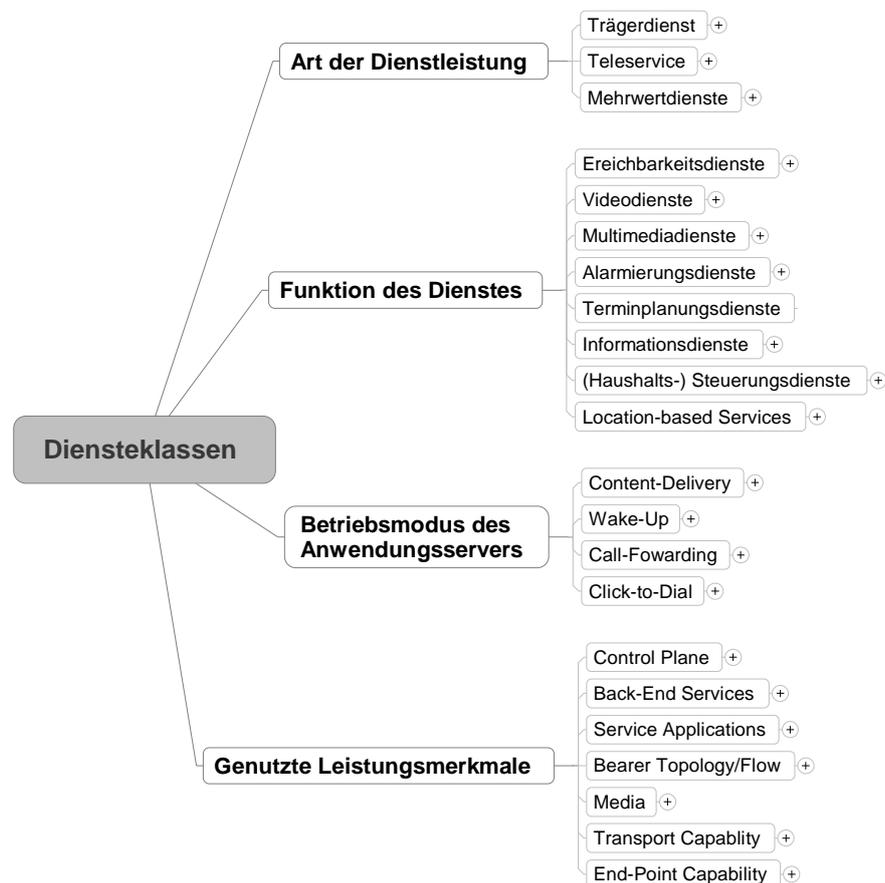


Abbildung 4.2: Klassifizierung von Diensten im NGN nach verschiedenen Kriterien (eine detaillierte Auflistung ist in Abb. B.1 zu finden).

4.2.1 Klassifizierung nach Art der Dienstleistung

Eine grundlegende Einteilung von Diensten ermöglicht die Art der Dienstleistung des Diensteanbieters gegenüber dem Nutzer [LeTO07a]. Diese Einteilung erweitert die gängige Unterteilung von Dienstleistung aus dem PSTN für das NGN. Dabei können im NGN folgende Dienstleistungen unterschieden werden:

Bearer Service als einfacher strukturierter Trägerdienst (z.B. für eine Sprachverbindung mit $64kBit/s$ und $8kHz$).

Teleservices z.B. Telefon- oder Telefax-Dienst (auf Basis des Bearer Service)

Mehrwertdienste Value-Added-Services, z.B. Televoting oder Click-to-Dial als erweiterte Anwendung, die über Teleservices hinausgehende Funktionalitäten bieten

Hinzu kommt, dass durch die Trennung von Datentransport und Diensten neue Arten von Netz- bzw. Diensteanbieter auftreten können. Im NGN sind damit drei verschiedene Arten von Anbietern möglich [LeTO08] (Abb. 4.3).

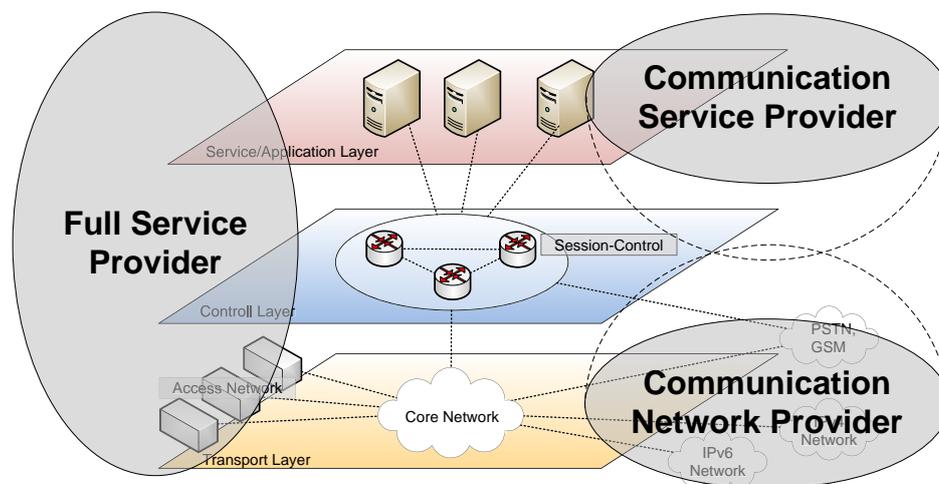


Abbildung 4.3: Im NGN können verschiedene Anbietertypen auftreten, die sich nur noch auf einzelne Schichten des NGN konzentrieren, statt auf einzelne Netztypen.

Ein **Communication Network Provider** (Zugangsanbieter) stellt mehrere Kern- oder Zugangsnetze bereit und ist somit primär für den Datentransport zuständig. Im NGN mit IP-Netzen oder Zugangsnetzen wie xDSL kann er sich auch auf Dienstleistungen der Transportschicht beschränken, bei anderen Telekommunikationsnetzen wie ISDN oder GSM muss er zusätzlich Vermittlungsdienste bereitstellen.

Der **Communication Service Provider** (Diensteanbieter) stellt Mehrwertdienste über eine Service Delivery Platform (SDP, siehe Kapitel 5.1.1) bereit. Diese Dienste sind hauptsächlich der Anwendungs- und Serviceschicht zuzuordnen, er kann aber auch Vermittlungsdienste anbieten.

Beide Bereiche deckt der **Full Service Provider** (Vollanbieter) ab. Er besitzt eigene Kern- und Zugangsnetze und Netzelemente in allen Schichten und bietet eigene Mehrwertdienste und Vermittlungsdienste an. Durch die vertikale Trennung der herkömmlichen Netze (Lagsagne vs. Spaghetti, s. Kap. 2.4) sind die meisten größeren Provider heute zwangsläufig Full Service Provider.

Durch die neue Möglichkeit eines reinen Zugangsanbieters, der sein Netz nur für den Datentransport zur Verfügung stellen kann und die Vermittlungs- und Mehrwertdienste von anderen Anbietern beziehen, sind völlig neue Szenarien vorstellbar. So könnten die lokalen Stadtwerke, die das lokale Stromnetz zum Datentransport nutzen, die Dienstleistung für Teleservices von einem anderen Anbieter zukaufen und dem Kunden gegenüber als Diensteanbieter für Telefonie auftreten.

4.2.2 Klassifizierung nach Funktion des Dienstes

Im Projekt „Dienste und Architekturen in zukünftigen Telekommunikationsnetzen“ (DazTel) der FH Frankfurt am Main wurde eine Unterscheidung von Mehrwertdiensten im NGN nach ihrer Funktion entworfen [BuTr07]. Dabei wurden verschiedene prototypische Dienste und ihre Nutzerakzeptanz untersucht, woraus sich folgende Funktionsgruppen ergeben haben:

Erreichbarkeitsdienste zur Erhöhung der Erreichbarkeit eines Nutzers, etwa durch Nutzung verschiedener Kanäle oder Gruppenanrufe

Multimedien Dienste zur Nutzung von mehreren Medienströmen, etwa für Videokonferenzen oder Telekooperation

Informationsdienste zur Übermittlung von relevanten Informationen an den Nutzer, etwa durch Abfrage von Online-Datenbanken oder Webservices

(Haushalts-) Steuerungsdienste zur Remote-Steuerung und Verwaltung von Maschinen oder automatisierten Haushalten

Alarmierungsdienste benachrichtigen den Nutzer beim Eintritt von bestimmten Ereignissen, wie Maschinenausfälle oder dem Ergebnis von Sportveranstaltungen

Terminplanungsdienste zur Unterstützung der Terminplanung, etwa mit Memo oder Notizfunktionen

Videodienste für die Übertragung von Videos, z.B. für IPTV

Standortbasierende Dienste die die Nutzerposition einbeziehen, beispielsweise zur Suche der nächsten Apotheke

Einfache Dienste, wie die in [BuTr07] vorgestellten prototypischen Dienste, lassen sich jeweils einer Gruppe zuordnen. Bei der Entwicklung komplexerer Dienste mit größerem Funktionsumfang oder durch die Einbeziehung von weiteren Diensten ist jedoch eine Mehrfachzuordnung nötig.

4.2.3 Klassifizierung nach Betriebsmodus des Anwendungsservers

Eine weiteres Klassifikationsmerkmal ist der Betriebsmodus des Anwendungsservers (AS), welcher den Dienst bereitstellt. Das bedeutet, dass die Klassifikation des Dienstes über die Rolle des Anwendungsservers als Netzelemente (vgl. Kapitel 3.3) im Kommunikationsablauf mit dem Nutzer bei der Dienststeuerung erfolgt. Prinzipiell bestehen für einen Anwendungsserver im NGN vier Möglichkeiten, einen Dienst bereitzustellen [LeTO07a].

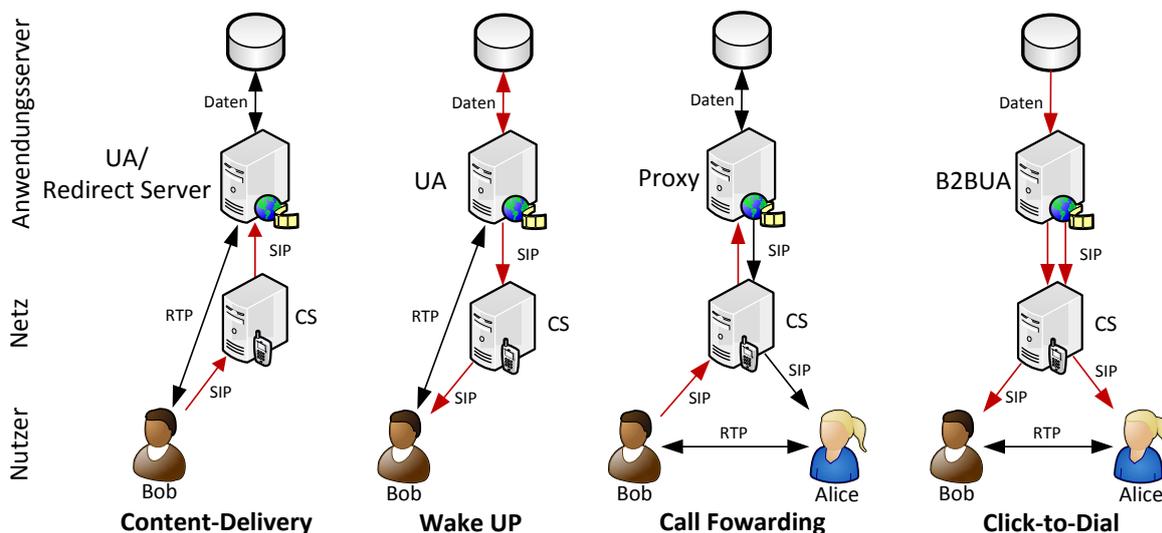


Abbildung 4.4: Betriebsmodi eines Anwendungsservers bei der Dienststeuerung, der Pfad zur Aktivierung des Dienstes ist rot markiert (frei nach [LeTO07a]).

Content-Delivery Der Nutzer löst den Dienst aus, indem er beispielsweise eine SIP-Nachricht an den AS sendet. Der AS funktioniert dabei als *UA* oder *Redirect Server* und sendet dem Nutzer anschließend die gewünschten Inhalte.

Wake-Up Der AS löst den Dienst selbständig aufgrund vorgegebener Parameter aus. Diese Parameter sind durch die Datenseite (Timer, Events) definiert. Anschließend erzeugt der AS als *UA* eine Verbindung zum Nutzer.

Call-Fowarding Der Nutzer kontaktiert den Anwendungsserver, welcher nach verschiedenen Parametern eine Verbindung zu einem anderen Nutzer auswählt und etabliert. Die Auswahl kann dabei nach vorher definierten Parametern (Uhrzeit, Nutzerstandort etc.) erfolgen. Der AS funktioniert dabei als *Proxy Server* für die Verbindung.

Click-to-Dial Der AS erhält beispielsweise über einen Webserver den Befehl eine Verbindung zwischen dem Nutzer und einem anderen Teilnehmer zu etablieren. Dafür leitet er als 3rd Party einen Anruf zwischen dem Endgerät des Nutzers und dem gewünschten Teilnehmer ein. Der AS ist in diesem Fall ein *Back-to-Back UA*.

Eine detaillierte Erläuterung des technischen Hintergrunds der Betriebsmodi ist in Kapitel 5.1.4 zu finden.

4.2.4 Klassifizierung nach genutzten Leistungsmerkmalen des Netzes

In [ETSI181004, S.15 ff.] wird vorgeschlagen, Dienste im NGN über die genutzten Leistungsmerkmale des Netzes zu gliedern. Dabei wird für jedes beteiligte Netzelement festgelegt welche Leistungsmerkmale genutzt werden.

Control Plane stellt Vermittlung und Steuerung von Verbindungen bereit, abhängig von den unterstützten Nachrichtentypen wie MMS, CHAT, IM oder Telefonie.

Back-End Services stellen für den Nutzer unsichtbare Dienstleistungen für die Erbringung der Basisdienste (Vermittlungsdienste, Nachrichtenübertragung) im Hintergrund bereit, etwa eine Routing-Datenbank oder verschiedene Kommunikationsdienste.

Service Applications sind netzinterne Dienste wie Nutzerauthentifizierung, Verbindungsvermittlung, Abrechnung oder Präsenzdienste.

Bearer Topology Flow sind die Trägerdienste für den Transport der verschiedenen Kommunikationsarten, wie Telefonie, Telefonkonferenzen, Multimedia, IM-, CHAT- oder MMS-Nachrichten.

Media bezeichnet alle benötigten Techniken zur Übermittlung von verschiedenen Datenströmen, etwa uni- oder bidirektionales Audiostreaming, Verschlüsselung von Mediendatenströmen, Media-Gateways oder Konferenzknoten.

Transport Capabilities ist die Übertragungsart der Daten, abhängig von der verwendeten Netzwerktechnologie. Im Internet und den meisten IP-basierenden Datenetzen kann nur Best Effort bereitgestellt werden. Die NGN-Kernnetze sollten aber auch Datenverbindungen mit QoS-Garantien bereitstellen können.

End-Point Capabilities sind Fähigkeiten des Nutzerendgerätes, wie die Übertragung von Audio, Video, MMS, CHAT, die Terminierung von verschiedenen Protokollen, Verschlüsselung oder die Bearbeitung des Nutzerprofiles auf dem Server des Zugangsanbieters.

Anhand dieser Kriterien lässt sich sehr einfach untersuchen, ob der Dienst im Netz angeboten werden kann. Für den Nutzer lässt sich so auch festlegen, welche Fähigkeiten das Endgerät besitzen muss, um den Dienst zu nutzen. Dabei sollte von der Seite des Anbieters aber bedacht werden, wie flexibel der Dienst bezüglich der Fähigkeiten des Endgerätes entwickelt wird, da sonst u.U. ein Großteil der möglichen Nutzer (und damit zahlenden Kunden) ausgesperrt werden könnte.

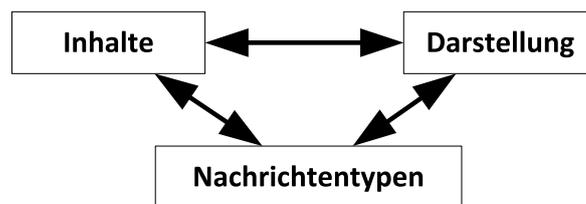


Abbildung 4.5: Bei der Bereitstellung eines Dienstes müssen drei Faktoren beachtet werden. Die Inhalte, welche zu transportieren sind, die möglichen Datentypen/Medien die verwendet werden können und die Darstellung auf dem Endgerät des Nutzers.

Durch die große Vielfalt auf dem Markt für Telefone und Mobiltelefone und der damit verbundenen Funktionsvielfalt stellt die Bereitstellung von Diensten, die auf allen Endgeräten laufen, eine sehr große Herausforderung dar. Dazu kommt, dass heute viele Endgeräte bereits selbst Anwendungen ausführen können um auf bestimmte Dienstleistungen zuzugreifen (siehe Kapitel 5.4). Damit wird teilweise schon den möglichen

Diensten im NGN vorgegriffen, da vergleichbare Dienste über eine Schnittstelle zum Internet von den Anwendungen in den Endgeräten selbst erzeugt werden.

4.3 Mehrwertdienste im NGN

Zur Einordnung der untersuchten Dienste im NGN bietet sich die Unterteilung nach *Funktion der Dienstleistung* (Kapitel 4.2.2) an, da aus Sicht des Nutzers die gebotene Funktionalität der Mehrwertdienste im Vordergrund steht und nicht deren Realisierung. In der Unterteilung aus Kapitel 4.2.2 ist jedoch keine Dienstklasse für die Haltung persönlicher Informationen, wie Kontaktdaten oder Memos vorhanden. Aus diesem Grund wird die Gruppe „Terminplanungsdienste“ auf alle PIM-Dienste (Personal Information Manager, siehe Kapitel 4.3.5) ausgeweitet.

Nachfolgend wird ein Überblick über Beispiele für Dienste der einzelnen Klassen gegeben. Die Ideen der Dienste stammen zu großen Teilen aus Whitepapers des Standardisierungsprozesses zum NGN. Die Konzepte zur Umsetzung der Dienste wurden im Rahmen dieser Arbeit entworfen.

4.3.1 Erreichbarkeitsdienste

Dienste, welche die Erreichbarkeit der Nutzer verbessern, sind nach [BuTr07] aus der Sicht der Nutzer heute die interessanteste Gruppe von Diensten, da der praktische Nutzen auf der Hand liegt. Bei den heutigen Kommunikationsformen steht das Endgerät im Mittelpunkt, da man nicht einen Nutzer, sondern ein Endgerät oder einen Anschluss in einem Zugangsnetz anruft. Durch die Vielzahl von Zugangsnetzen, die im NGN zu einem Kommunikationszugang zusammengefasst werden (Mobilfunk, PSTN, DSL etc.), bietet das Netz bereits eine technische Grundlage für eine bessere Erreichbarkeit mit dem Nutzer als zentralem Element.

Device Unify Service (DUS)

Ein Ansatz, welcher den Nutzer zusätzlich von der Funktionalität der einzelnen Kommunikationsgeräte für die Zugangsnetze entkoppelt, ist der „*Device Unify Service*“ [ETSI181003, S.15]. Dieser Dienst soll aus Sicht des Nutzers alle Endgeräte vereinheitlichen, so dass sie sich für ihn wie ein virtuelles Gerät mit mehreren Eingabe- und Ausgabemöglichkeiten verhalten. Damit wäre es für den Nutzer egal, welches Endgerät und welches Zugangsnetz er gerade verwendet und hätte eine „*Always-On*“-Funktionalität. Besonders wichtig ist dabei, dass sich die Funktionalitäten der Geräte und die verfüg-

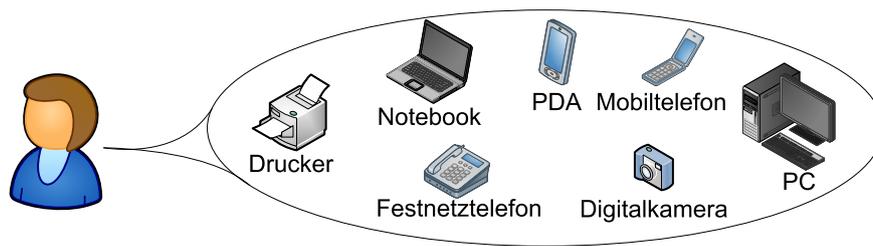


Abbildung 4.6: Der „Device Unify Service“ fasst aus Sicht des Nutzers seine Geräte zu einem virtuellen Endgerät mit mehreren Ein- und Ausgabemöglichkeiten zusammen.

baren Daten (z.B. Kontaktliste) nicht unterscheiden, um eine einheitliche Benutzerführung und Bedienbarkeit zu erreichen. Die einzelnen Endgeräte haben unterschiedliche Terminalprofile, können also entsprechend ihres Bauart beispielsweise nur bestimmte Nachrichtentypen anzeigen und nicht alle Medienströme verarbeiten. Um die Erreichbarkeit des Nutzers sicherzustellen existieren zwei verschiedene Ansätze: Der Nutzer kann das Verhalten der Geräte selbst beeinflussen, um so ein statisches oder dynamisches Routing der Kommunikationsverbindungen zu erreichen. Andererseits kann diese Anpassung auch vollautomatisch erfolgen, so dass der Nutzer ohne jeglichen Aufwand (fast) immer erreichbar ist. Dieser Service bietet die in Kapitel 3.5.2 beschriebene Mobilität zwischen Netzen und Endgeräten.

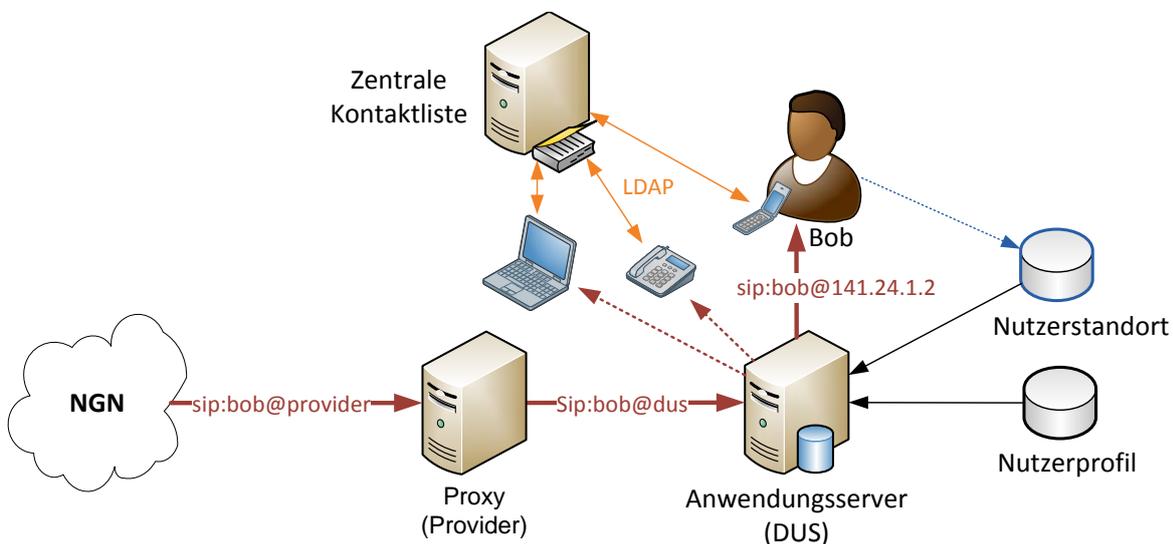


Abbildung 4.7: Für die Bereitstellung des Device-Unify-Services muss ein Anwendungsserver die Weiterleitung eingehender Sitzungen anhand von komplexen Regeln (z.B. auf Basis des Nutzerstandorts und -profils) übernehmen.

Für die Realisierung wird dabei ein Netzknoten benötigt, der Informationen über die

einzelnen Endgeräte, den Nutzerstandort und das Nutzerprofil besitzt, so dass er eingehende Verbindungen auf das richtige Endgerät weiterleitet. Der Nutzer ist weiterhin nur über eine einzige ständige URI erreichbar. Ein eingehender Anruf wird in diesem Fall vom Proxy Server nicht einfach an eine hinterlegte temporäre URI weitergeleitet, sondern ein angeschlossener Anwendungsserver muss entscheiden welches Endgerät des Nutzers für die Verbindung passend ist. In diese Entscheidung können der Inhalt der eingehenden Verbindung (Video-, Audio- oder Textdaten) oder der Nutzerstandort (im Auto oder im Wohnzimmer) eingehen. Die Auswertung solcher komplexen Profile ist für Proxy-Server nicht vorgesehen, da sie über das HSS nur eine feste Zuordnung zwischen temporären und dauerhaften URIs abfragen können. Für ausgehende Anrufe wird dabei keine Anpassung erforderlich, da die Verbindung dort über die aktuelle temporäre URI des Nutzers zustandekommt. Die zentrale Kontaktliste kann zwischen den Geräten über einen zentralen Server per LDAP ausgetauscht werden, das für die verteilte Abfrage von Verzeichnisdiensten ausgelegt ist.

Ring Back When Free (RBWF)

Im PSTN wird von einigen Anbieter eine automatische Wahlwiederholung angeboten, wenn die Zielrufnummer besetzt ist. Diese Funktion lässt sich im NGN zu „*Ring Back When Free*“ (RBWF) erweitern [jNet09], die spezielle Leistungsmerkmale des NGN nutzen kann. Im PSTN wird zur Ermittlung des Zielstatus einfach die Rufnummer des Ziels solange gewählt, bis kein Besetztzeichen mehr erscheint. Anschließend wird der Initiator angerufen und eine Verbindung hergestellt.

Im NGN kann dieser Dienst wesentlich effektiver realisiert werden, ohne zur Statusabfrage einen vollen Sprachkanal zu belegen. So kann zur Ermittlung der Gesprächsbereitschaft des Gesprächspartner auf den Präsenzdienst (siehe Kapitel 3.3.8) zugegriffen werden. Über diesen kann durch die SIP-Nachrichten SUBSCRIBE und NOTIFY abgefragt werden, ob der Gesprächspartner ein Gespräch annehmen kann (Präsenz-Status *open*) oder nicht verfügbar ist (Präsenz-Status *close*) [RFC3863]. Dieser Erreichbarkeitsstatus kann für jedes Medium (Audio, Video etc.) des UAs einzeln definiert werden.

Die erweiterten Statusfelder können dafür genutzt werden, Gespräche nur bei einer gewissen Dringlichkeit durchzuführen und Rücksicht auf die aktuelle Tätigkeit des Gesprächspartners zu nehmen. Solche Zustände könnten beispielsweise *Dienstberatung*, *Nutzer im Auto*, *Mittagspause* oder *Freizeit* sein.

Die Benachrichtigung des Dienstes an den Nutzer über einen veränderten Status des Gesprächspartners sollte im einfachsten Fall als Anruf beim einleitenden Gesprächspartner erfolgen. Nach der Mitteilung über die Änderung des Status, durch eine NOTIFY-

Nachricht des Presence Servers, kann der Call Server den Nutzer anrufen, und nach der Annahme des Rückrufes das Gespräch zum Gesprächspartner einleiten. Unterstützt das Endgerät auch noch andere Nachrichtentypen (z.B. MMS), so kann der Anrufer auch einfach eine Benachrichtigung bekommen, dass der gewünschte Gesprächspartner jetzt wieder verfügbar ist und zu einem selbstgewählten Zeitpunkt von sich aus anrufen.

Der Dienst kann entweder als Funktion direkt im UA implementiert sein, der nach einem erfolglosen Verbindungsversuch die Status des Gesprächspartners überwacht, oder er muss von einem Netzknoten ausgelöst werden. Eine reine Implementierung auf einem Anwendungsserver ist hier nicht möglich, da der Dienst automatisch aktiviert werden muss, sobald eine Verbindung nicht hergestellt werden kann und als SIP-Status *486 Busy Here* erhält. Der Anwendungsserver muss nach dieser Statusmeldung aktiv vom Netzknoten (z.B. Proxy-Server) benachrichtigt werden.

Push-to-talk over Cellular

Der Dienst *Push-to-talk over Cellular* (PoC) [3GPP23979] ermöglicht Gruppenanrufe per Knopfdruck, ähnlich der Funktion eines Walkie-Talkies. Dabei wird eine Taste am Endgerät gedrückt, und anschließend eine Sprachnachricht von einem Teilnehmer an einen vorher definierten Empfängerkreis gesendet. Möchte einer der Empfänger antworten, kann dieser ebenfalls mit einem Tastendruck eine Sprachnachricht an alle beteiligten senden. Dies kann für die direkte Kommunikation einer Gruppe genutzt werden, etwa für Großveranstaltungen, innerbetriebliche Abläufe oder Gruppenchats.

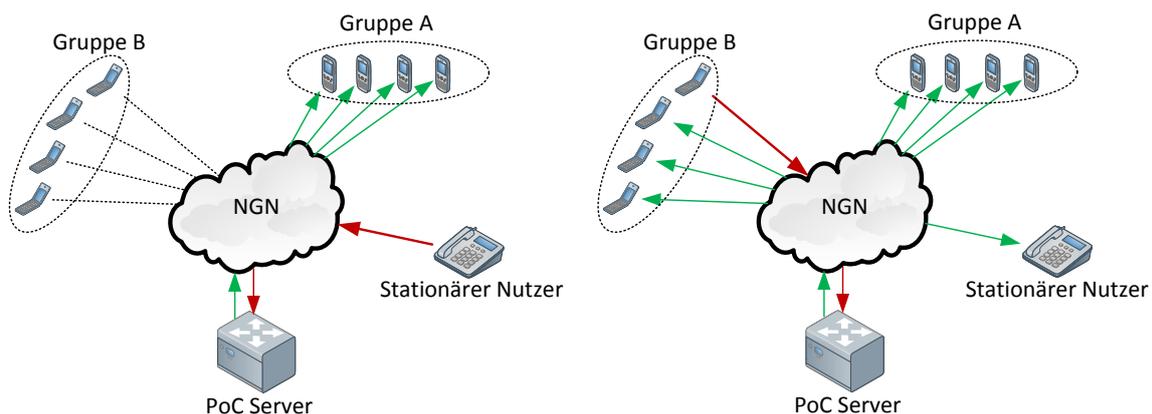


Abbildung 4.8: Push-to-talk over Cellular (PoC) ermöglicht die Funktionalität von Handfunkgeräten mit selektiven Gruppenrufen

Dazu wird die Nachricht eines Nutzers über das Netz zum PoC-Server übertragen, der diese Nachricht anschließend per Multicast an den vorher definierten Empfänger-

kreis versendet. Da dieser dann in einem Multicast-Baum eingetragen ist, muss die Nachricht vom Server nur ein einziges Mal an eine Multicast-Adresse übertragen werden.

Ein großer Vorteil gegenüber herkömmlichen Funkgeräten ist, dass die Reichweite nicht durch die Endgeräte selbst begrenzt wird und im gesamten NGN funktioniert. So können auch stationäre oder mobile Geräte an anderen weit entfernten Standorten in die Kommunikation einbezogen werden (z.B. Koordinator im Lagezentrum). Weiterhin lassen sich so auch verschiedene Empfängerkreise definieren und per Knopfdruck direkt ansprechen (z.B. auf einer Veranstaltung: nur Sicherheitspersonal, nur Reinigungspersonal oder Bühnentechniker). Prinzipiell ist dieser Dienst auch eingeschränkt in herkömmlichen Netzen implementierbar, wobei man hier aufgrund der Abrechnung pro Verbindung (Pay-per-Call) und der begrenzten Signalisierung sehr eingegrenzt ist. Durch die fehlende Unterstützung von Multicast kommt bei großen Empfängergruppen hinzu, dass der Server die Nachricht an jeden Nutzer einzeln versenden muss und damit eine hohe Netzwerklast verursacht. Im NGN besteht zudem die Möglichkeit, Nachrichten über einen beliebigen Anwendungsserver oder Proxy direkt an die Nutzer zu übertragen, ohne dass die angesprochenen Nutzer jede Verbindung manuell annehmen müssen, da Nachrichten auch verbindungslos übertragen werden können. Der Dienst kann auch auf einem eigenen Anwendungsserver implementiert werden, was eine zentrale Gruppensteuerung mit Anbindung an eine interne Gruppenverwaltung einer Organisation (Domänencontroller, Nutzerkontensteuerung) ermöglicht.

Voice on Web

Ein User Agent auf einem PC ist meistens ein Programm, das lokal installiert wird (z.B. X-Lite). „Voice on Web“ geht den Weg, dass der UA als Plugin innerhalb einer Webseite eingebettet ist [dVel07, S.139]. Dieses Plugin kann dann geräteunabhängig und betriebssystemunabhängig genutzt werden, so dass beispielsweise in einem Internetcafé der eigene NGN-Anschluss genutzt werden kann. Ein solcher Dienst wird schon seit einigen Jahren von *Abbeyphone*² angeboten.

„Voice on Web“ bietet aber auch noch andere Anwendungsmöglichkeiten. So können Unternehmen, die eine zusätzliche Kontaktmöglichkeit neben E-Mail-Formularen, Live-Chats oder Telefonhotlines wünschen, ein *Voice-on-Web-Plugin* auf ihrer Webseite anbieten, worüber der Nutzer kostenlos direkt mit Supportmitarbeitern sprechen kann. Für den Nutzer entspricht die Funktionalität weitestgehend Click-to-Dial, nur

²<http://www.abbeyphone.com/>

das er hier keinen gesonderten UA (Softphone oder IP-Phone) mehr benötigt. Damit sind auch anonyme Angebote, wie beispielsweise Suchtberatung möglich, da der Nutzer nicht von seinem eigenen NGN-Anschluss mit einer zurückverfolgbaren Kennung anrufen muss.

Das Plugin kann auf einer Webseite beispielsweise als Java-, Flash- oder als ActiveX-Plugin zur Verfügung gestellt werden. Es muss Zugriff auf die lokalen Ein- und Ausgabegeräte für Audio erhalten. Das Plugin selbst überträgt nur die Audiodaten zum Anwendungsserver, welcher dann letztendlich selbst eine SIP-Verbindung etabliert und die Audiodaten darüber weiterleitet.

Die Funktionalität kann zwar auch außerhalb eines NGN realisiert werden, innerhalb des NGN kann jedoch auf die bestehende Verbindungssteuerung und die QoS-Techniken zurückgegriffen werden.

Temporäre URI

Durch die Nutzung eines IP-basierenden Kernnetzes wird es möglich, mit geringem Aufwand ein automatisiertes Anrufsystem im Netz zu betreiben, womit kostengünstig automatisiert eine Vielzahl von Werbeanrufen abgewickelt werden können. Diese Anrufe werden Analog zu unerwünschter Werbung via E-Mail als *Spam via Internet Telephony* (SPIT) bezeichnet [Detk07, S.104].

Bei Werbung via E-Mail hat sich als beste Abwehrmethode die kontrollierte Verbreitung (Geheimhaltung) der Mailadresse herausgestellt. Für die Kommunikation mit weniger vertrauenswürdigen Stellen werden Einweg-Emailadressen angeboten, die manuell oder nach Ablauf einer bestimmten Frist verfallen. Damit ist es unerheblich, was mit der Adresse nach Abschluss der Kommunikation passiert.

Da praktisch jeder Server im NGN mit einem DNS-Eintrag oder einer eigenen IP-Adresse beliebige URIs bereitstellen kann, bietet sich hier die Einrichtung von temporäre *Einweg-URIs* an [jNet09]. Diese Rufnummer URI kann bei Transaktionen oder Kontakten verwendet werden, bei denen unklar ist, wer anschließend auf die Kontaktdaten zugreifen kann.

Die Gespräche kann der Server, dem die URI zugeordnet ist, als Proxy Server weiterleiten. Da im NGN nur die Vermittlungsdaten für den Gesprächsaufbau übertragen werden müssen, können die Gesprächsdaten nach dem Verbindungsaufbau direkt zwischen den Gesprächspartnern ausgetauscht werden, was für den Server eine Entlastung darstellt. Bei Einweg-Mail-Adressen muss der bereitstellende Server dagegen alle eingehenden Nachrichten vollständig empfangen und entsprechend weiterleiten. Die Funktionalität des Servers ist die eines kombinierten Registrar/Proxy-Servers mit

dem Unterschied, dass die zugewiesenen dauerhaften SIP-URIs nicht dauerhaft genutzt werden.

Für den Nutzer erhöht sich damit seine eigene Erreichbarkeit, da er seine persönliche URI effektiv vor SPIT schützen und die temporären URIs je nach Bedarf löschen oder neu erzeugen kann.

4.3.2 Videostreamingdienste

Die Nutzung des NGN für Videostreamingdienste sind bereits seit Anfang an ein fester Bestandteil der Entwicklung von IMS gewesen [Drlj09, TrWe07].

IPTV

Die Liveübertragung von herkömmlichen Fernsehen über IP-Netze als Videostream bezeichnet man als *IPTV*. Die Übertragung über Datennetze bietet dabei für die Sendeanstalten erst mal nur einen weiteren Übertragungsweg, zusätzlich zu Fernsehen via Satellit oder Kabelfernsehen.

Das Datennetz muss dafür Multicasting unterstützen, damit für die Übertragung bis zu den Nutzern die Bandbreite effektiv genutzt wird. Von Vorteil gegenüber herkömmlichen Verbreitungswegen für Fernsehen ist der deutlich flexiblere Zugang zu den Netzen als durch die bisherigen Kabel- und Satellitenbetreiber. Heute werden die Fernsehnetze von großen Medienkonzernen betrieben, und durch die Knappheit der Kabelfrequenzen und Satellitentransponder erhalten meist nur große Fernsehsender Zugang zu den Verbreitungswegen. Durch den Wegfall der starren Strukturen besteht so auch die Möglichkeit, dass auch kleinere private und lokale Fernsehsender ihre Dienste anbieten können. Es besteht die Möglichkeit, dass sich so eine vergleichbare Sendervielfalt ausbilden kann, wie bei den Internetradiostationen.

IPTV ist für die Verbreitung aber nicht der einzige Weg, da es besonders im lokalen Raum, durch die Digitalisierung des terrestrischen Fernsehens möglich ist einen DVB-T-Kanal zu erhalten. Der Vorteil von IPTV ist hier aber eindeutig die Reichweite, so dass auch Sender für kleine Zielgruppen landesweit mit wenig Aufwand senden kann.

Video-on-Demand

Ein interessanter neuer Dienst für die Endnutzer ist *Video-on-Demand* (VoD), der Abruf von Videos von einem Server zu einem beliebigen Zeitpunkt. Damit können Pay-per-View-Modelle realisiert werden, bei welchen der Nutzer wie in einer Videothek einen Film „entleiht“, und anschließend das Recht erhält, diesen für einen bestimmten

Zeitraum vom Server abzurufen. Kann er diese Wiedergabe bidirektional auf dem Server steuern, wird dies als *Interactive Video-on-Demand* bezeichnet [ETSI181003, S.53]. Es besteht aber auch die Möglichkeit, den Film per Streaming in den Speicher der Set-Top-Box des Nutzers zu übertragen, so dass er die Wiedergabe lokal beeinflussen kann, ohne dass der Server weitere Steuersignale empfängt.

Die öffentlich-rechtlichen Fernsehanstalten in Deutschland bieten sogar eigene Portale an, bei denen der Nutzer einen großen Teil der Sendungen nachträglich kostenlos abrufen kann. Ein bekanntes Beispiel ist die ZDF-Mediathek³, in welcher fast alle Sendungen, die das ZDF gesendet hat, ungefähr 13 Monate lang vorgehalten. Es existieren weiterhin eine Reihe kommerzieller Portale, etwa von Apple Inc. für die Set-Top-Box AppleTV oder von Microsoft für die Spielekonsole Xbox 360. Besonders die Erweiterung des bereits stark genutzten Online-Musikladens Apple iTunes zu einem On-Demand-Streaming-Portal verspricht für den Betreiber ein attraktiver Dienst zu werden, da vom Start weg bereits eine hohe Anzahl an Nutzern vorhanden ist.

Bei der Nutzung über herkömmliche IP-Netze besteht hier allerdings das Problem, dass sich bei einer großen Nutzerzahl oder starkem Datenverkehr die Wiedergabequalität deutlich verschlechtert, da die meisten IP-Netze keine QoS-Garantien ermöglichen. Bei VoD wird dieses Problem häufig durch einen Pufferspeicher gelöst, so dass die Videowiedergabe verzögert gestartet wird und Kapazitätsengpässe über den Puffer abgefangen werden können. Microsoft hat für das Streaming von hochauflösendem Videomaterial ohne Wartezeit eine Lösung vorgestellt, bei welcher das Video beim Beginn der Wiedergabe in niedrigerer Auflösung abgespielt wird, und sobald der Puffer gefüllt ist das hochauflösende Videomaterial wiedergegeben wird [Gies09]. Damit ist auch ein fast verzögerungsfreies Vor- und Zurückspulen möglich, um einen ähnlichen Bedienkomfort wie bei einer Wiedergabe einer lokalen Videodatei oder einer DVD zu ermöglichen.

Residential Broadband Entertainment

Aktuell bieten IPTV-Empfänger nicht viel mehr als etwa digitales Fernsehen via Satellit oder Kabel. Nur ein „echter“⁴ Video-on-Demand-Dienst bietet für den Nutzer einen wirklichen Mehrwert gegenüber herkömmlichem Fernsehen, abgesehen davon, dass er

³<http://www.zdf.de/ZDFmediathek/>

⁴Der Fernsehsender Sky (vormals Premiere) hat mit seinem Select-Angebot bereits vor längerer Zeit ein ähnliches Angebot mit beschränkter Auswahl und festen Sendezeiten realisiert. Der Nutzer hat dabei die Wiedergabeberechtigung für einen Film erworben der auf speziellen PayTV-Fernsehkänen mehrmals täglich gesendet wurde und zeitverzögert an seiner Set-Top-Box abgespielt werden konnte.

nur noch einen Internetanschluss mit entsprechender Bandbreite benötigt, statt einen separaten Anschluss für jeden Dienst. Zusätzlich zur Übertragung von herkömmlichen Fernsehen bietet das NGN durch die bidirektionale Übertragung die Möglichkeit für interaktives Fernsehen, etwa für Votingdienste per Set-Top-Box. Solche Dienste können mit herkömmlichen Medien (Radio/TV), Internetdiensten (Internetradio/Informationsdienste) und On-Demand-Diensten in einem „Residential Broadband Entertainment“-Portal gebündelt werden [ETSI181003].



Abbildung 4.9: Set-Top-Box für das Breitbandportal der Deutschen Telekom zur Nutzung von IPTV und Video-on-Demand (*Quelle*⁵)

Angebote für Triple-Play (Telefon, IPTV, Internetzugang) über IP-Netze sind in Deutschland bereits seit 2006 verfügbar und im September 2009 nutzten bereits 800.000 Kunden der Deutschen Telekom AG dieses Angebot [Brie09b], das bei ungefähr 20 Mio. Haushalten in Deutschland einen Marktanteil von 4% bedeutet.

Dabei kommt eine Set-Top-Box (Abb. 4.9) zum Einsatz, über welche der Nutzer unter einer einheitlichen Oberfläche Fernsehen (IPTV mit Timeshift und Aufnahme-funktion), Internetradio und eine umfangreiche Auswahl an abrufbaren Filmen nutzen kann. Eine Nutzung über andere Endgeräte (PC, Smartphones oder Set-Top-Boxen anderer Hersteller) oder die Einbindung von Telefonie ist aktuell jedoch nicht möglich oder vorgesehen. Da dieses Portal jedoch erst in der Set-Top-Box gebündelt wird, müssen IPTV und VoD keine gemeinsame Plattform darstellen. Für die Nutzung mit anderen Endgeräten ist eine solche gemeinsame Plattform jedoch von Vorteil, da man sonst wieder auf zwei getrennte Dienste zugreifen muss.

⁵Deutsche Telekom: Pressebilder <http://www.telekom.com/dtag/cms/content/dt/de/89884>

4.3.3 Multimediadienste

Multimediaanwendungen wie Videotelefonie, Videokonferenzen oder Telekooperation [BuTr07][ETSI181003] stellen eine der größten Neuerungen dar, weil sie durch die Einführung von neuen Medien das Potential haben, die Kommunikation zwischen den Nutzern ähnlich grundlegend zu beeinflussen, wie die Einführung der Mobilkommunikation. Da unter Multimedia alle Dienste fallen, die zur Erfüllung ihrer Aufgabe mehrere Medienströme kombinieren, umfasst diese Klasse ein sehr breitgefächertes Feld von Anwendungen.

Telefon-Streamingdienste

Ein relativ einfacher Dienst, der auch mit herkömmliche Endgeräten per Gateway im NGN genutzt werden kann, ist ein Audiostreamingdienst. Dabei kann der Nutzer per SIP-Nachricht, Sprachwahl oder DTMF zwischen verschiedenen Streams auswählen und anschließend entsprechend abhören [BuTr07]. Dieser Dienst kann beispielsweise für die Liveübertragung von Konferenzen, Sportereignisse oder Radiosender genutzt werden. Weiterhin ist auch ein On-Demand-Abruf von aktuellen Nachrichten ebenfalls denkbar.

Bei der Liveübertragung von Veranstaltungen wird dabei nur ein Mediaserver benötigt, der den Quelldatenstrom in ein kompatibles Format für den UA umwandelt. Die Sitzung etabliert der Nutzer dabei zwischen seinem UA und einem Anwendungsserver, der anschließend einen Mediendatenstrom zwischen dem Mediaserver und dem UA einleitet. Die Aktivierung des Dienstes kann dabei durch den Aufruf einer speziellen SIP-URI erfolgen (z.B. `sip:tagesthemen@anwendungsserver`), die bereits die gewünschte Dienstleistung (Liveübertragung der Nachrichtensendung *Tagesthemen*) spezifiziert. Da in der INVITE-Nachricht des SIP-Protokolls Informationen zu den unterstützten Medien enthalten sind, kann der Anwendungsserver daraus entnehmen, welche Medienströme das Endgerät verarbeiten kann. Mit einem reinen Telefon ohne Videofunktion würde man auf diesem Weg nur einen Audiodatenstrom erhalten. Für den Abruf von gespeicherten Audiodatenströmen wird zusätzlich ein Nutzerinterface, beispielsweise ein sprachgesteuertes Menü, zur Auswahl des gewünschten Streams benötigt. Der Anwendungsserver kann dann erst durch die Auswahl des Nutzers einen Datenstrom einleiten.

IPTV Call Notifier

Während der Wiedergabe des IPTV-Dienstes mit einer Set-Top-Box, kann über den Dienst *IPTV Call Notifier* eine Mitteilung über einen eingehenden Anruf auf einem der anderen Endgeräte des Nutzers in die aktuelle Wiedergabe eingeblendet werden [jNet09].

Für die Realisierung dieses Dienstes sind mehrere Möglichkeiten denkbar. Soll der Dienst ohne spezielle Anpassungen auf allen IPTV-fähigen Geräten funktionieren, kann der Dienst auf einem Mediaserver implementiert werden. Dazu muss der Server die Videodaten manipulieren und dem Videodatenstrom einen Hinweis auf den eingehenden Anruf hinzufügen, bevor er sie an den entsprechenden Empfänger weiterleitet. Dieses Verfahren funktioniert jedoch nicht, wenn der IPTV-Empfänger eine Aufnahme-funktion (Timeshift oder Videorekorder) besitzt, weil die Meldung dann möglicherweise nicht zum vorgesehenen Zeitpunkt angezeigt wird. Außerdem läuft die gezielte Manipulation des Videostreams für genau einen Teilnehmer dem Grundgedanken von Multicast zuwider, nur einen Videodatenstrom für alle Teilnehmer zu versenden, welcher dann im Netz an die Empfänger verzweigt wird. Der Mediaserver muss dazu jederzeit in alle Videodatenströme zu verschiedenen Nutzern eingreifen können, um die Nachricht entsprechend einzubetten. Verwendet man jedoch einen lokalen Mediaserver, der etwa im Haushalt eine Video- und Musikbibliothek bereitstellt, kann dieser den ausgehenden Medienstrom für jeden Streaming-Client problemlos beeinflussen.

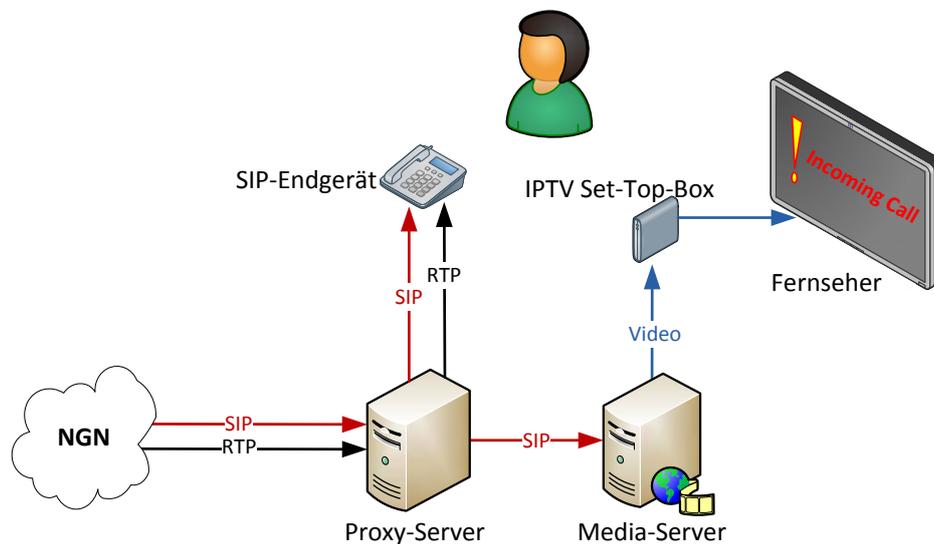


Abbildung 4.10: Ein IPTV-Call-Notifier blendet Informationen über eingehende Gespräche in ein laufendes Videostream ein. Dazu muss der Media-Server eine Benachrichtigung über eingehende Gespräche erhalten, etwa indem der Proxy diese Verbindungen auch an ihn weiterleitet.

Die praktikablere Lösung für die Verwendung in einem Multicast-Netz ist deshalb ein IPTV-Empfänger, welcher einfache SIP-Nachrichten empfangen kann, so dass er bei dem Empfang einer Benachrichtigung eine Einblendung auf dem Bildschirm erzeugt. Um diese Nachricht zu erzeugen, muss allerdings ein Netzelement (beispielsweise der Anwendungsserver) die Benachrichtigung über die eingehende Verbindung neben dem eigentlichen Endgerät an den IPTV-Empfänger senden. Da SIP aber auch die Registrierung von mehreren UAs beim Registrar Server unterstützt, kann sich der IPTV-Empfänger aber auch als UA für die ständige URI des Nutzers registrieren und erhält so bei einem eingehenden Gespräch ebenfalls eine INVITE-Nachricht. Hierfür muss aber sichergestellt sein, dass der Proxy-Server bei einer mehrfachen Registrierung für eine URI das Gespräch an alle registrierten Endsysteme weiterleitet.

Personal Contact Pages

Die Individualisierung des eigenen Kommunikationsanschlusses über *Ringback Caller Tones* wird in den USA bereits für die herkömmlichen Mobilfunknetze angeboten [Cris09]. Der Nutzer kann damit festlegen, welche Töne oder Audiodateien ein Anrufer anstatt des Freizeichens hört, bevor das Gespräch am eigenen Endgerät angenommen wird. Diese Serviceleistung ist nicht zu verwechseln mit Warteschleifen mit musikalischer Untermalung, wie sie heute auch eingesetzt werden, da diese Informationen erst nach dem Zustandekommen der Kommunikationsverbindung übertragen werden. *Ringback Caller Tones* werden also vom Netz als Leistungsmerkmal bereitgestellt, und sind endgeräteunabhängig.

Im Bereich der Multimediakommunikation im NGN lässt sich dieses Konzept auf *Personal Contact Pages* (Early Media) erweitern [dVel07, S.136]. Der Anrufer hört dann nicht mehr nur Audiodateien, sondern sieht ein Begrüßungsvideo oder erhält Informationen in Form von Text oder Bildern bevor der Anruf zustande kommt. So könnte etwa eine kurze Firmenpräsentation während der Wartezeit eingespielt werden, oder bei einer Arztpraxis die Sprechstundenzeiten schon vor dem Gespräch angezeigt werden (Abb. 4.11).

Dieser Dienst kann durch jede beliebige IP-Telefonanlage realisiert werden, wenn vorgefertigte Multimediadateien während der Wartezeit abgespielt werden sollen. Es können jedoch auch in Echtzeit generierte aktuelle Informationen, beispielsweise zu Sonderangeboten oder aktuellen Veranstaltungen, genutzt werden. Diese müssen dann jedoch von einem Anwendungsserver in Echtzeit als Videodatenstrom erzeugt und dann von der Telefonanlage eingespielt werden. Dazu wird eine eingehende Sitzung für einen beliebigen Teilnehmer an der lokalen Telefonanlage sofort angenommen und mit ei-



Abbildung 4.11: Personal Contact Pages können während des Anrufes zur Information des Anrufers genutzt werden, und dabei Daten wie Öffnungszeiten oder Optionen eines Sprachmenüs als Videostream auf seinem Endgerät anzeigen (Fotomontage).

ner Video- und Audioverbindung beantwortet, wobei der Nutzer dann beispielsweise weiterhin das Freizeichen als Audiostream hört und im Display bereits ein Videostream mit Informationen empfängt.

Multichannel Media

Die Tonqualität ist bei Voice-over-IP ein generelles Problem. Laufzeit- und bandbreitebedingte Qualitätseinbußen sollen im NGN durch QoS-Garantien abgefangen werden (siehe Kapite 3.5.1). Um die Tonqualität noch weiter zu verbessern wurde bereits ein neuer Codec mit 16kHz Abtastfrequenz eingeführt (G.722). Damit ist der nächste Schritt zu einer besseren Audiowiedergabe, vergleichbar mit dem Hifi-Bereich, hin zu einer Stereo- oder Mehrkanalübertragung. Da bei SIP-Verbindungen die Anzahl der möglichen parallelen Verbindungen nicht mehr durch die Anzahl der physikalischen Leitungen begrenzt wird, können mehrere Audiokanäle parallel übertragen werden.

Als Ergebnis erhält man Mehrkanalaudiodatenströme, beispielsweise Stereo, 5.1 oder beliebige Konfigurationen. Diese können durch mehrere Mikrofone in den Ecken eines

Raumes oder Mikrofonarrays erzeugt werden. Die Wiedergabe erfolgt anschließend über ein Mehrkanal-Audiosystem. Man kann die Informationen aber auch sinnvoll nutzen, wenn einer der Gesprächspartner nur ein Stereo-Headset nutzt. So kann über die Raumposition seines Headsets eine virtuelles Räumlichkeitsgefühl erzeugt werden [dVel07, S.137].

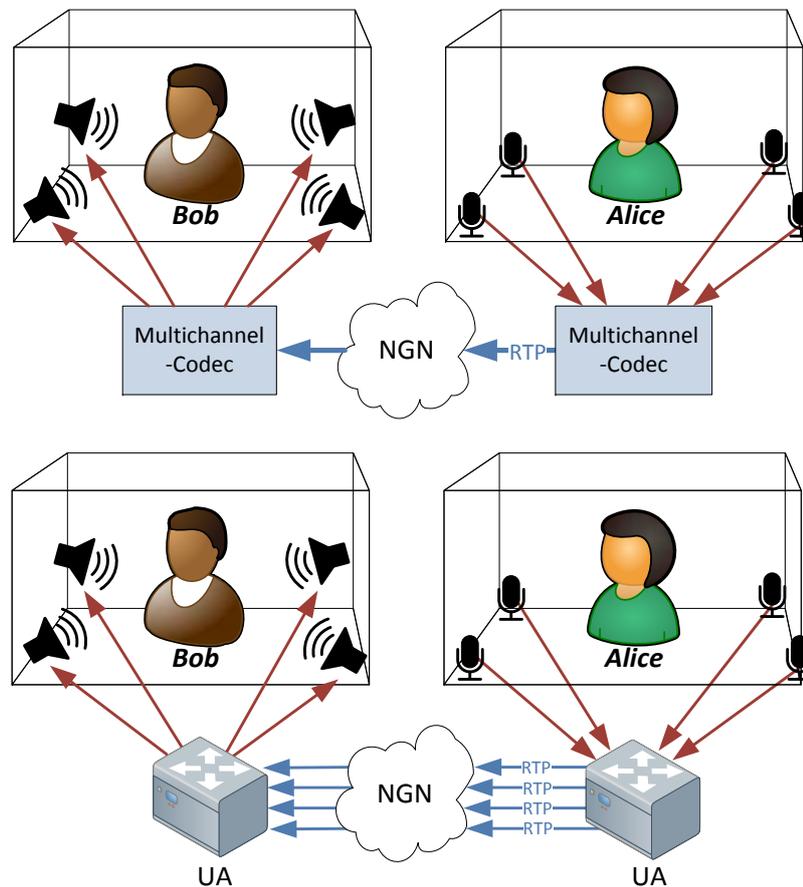


Abbildung 4.12: Mehrkanal-Sitzungen können über einen speziellen Multichannel-Codec oder über separate Datenströme mit herkömmlichen Codecs realisiert werden.

Zur Realisierung sind prinzipiell zwei Wege denkbar. Zum einen könnte ein spezieller echtzeitfähiger Mehrkanalcodec verwendet werden, vergleichbar mit Dolby Digital Live [AC3]. Dieser muss in beiden UAs verfügbar sein, und könnte direkt im SDP-Teil der SIP-INVITE-Nachricht beim Verbindungsaufbau referenziert werden. Wenn beide UAs diesen Codec unterstützen wird automatisch eine Mehrkanalverbindung aufgebaut, weitere Änderungen an den weiterleitenden Netzelementen sind nicht nötig, da die Sitzungssteuerung vollständig unabhängig von den verwendeten Codecs ist.

Weiterhin könnten mehrere parallele RTP-Verbindungen mit herkömmlichen Co-

decs genutzt werden. Dabei wäre es möglich Übertragungsfehler in den einzelnen Datenströmen durch die anderen Datenströme zu kompensieren, denn bei einer Störung von einem Stereokanal gehen nur Rauminformationen verloren, da ein großer Teil der Sprachinformationen redundant in beiden Kanälen transportiert wird. Die getrennte Übertragung erfordert aber eine exakte Synchronisation der einzelnen Kanäle auf der Empfängerseite, weil es sonst zu Echo-Effekten kommt. Die benötigte Bandbreite gegenüber einem speziellen Mehrkanalcodec ist ebenfalls deutlich höher, weil dieser in der Regel die Redundanz zwischen den Kanälen zur Verringerung der Datenmenge nutzt. Da die parallelen Verbindungen ebenfalls einen angepassten UA benötigen, da dieser mehrere synchronisierte Verbindungen bereistellen muss, ist die Realisierung über einen Mehrkanalcodec deutlich einfacher, da die Synchronisierung wegfällt, die Datenrate geringer ist und der Overhead der vielen gleichzeitigen Verbindungen entfällt.

Die Raumposition der Gesprächspartner kann dabei aus den Mehrkanalaudioströmen berechnet werden. Nutzt einer der Teilnehmer ein Monoheadset, muss dessen Position bestimmt (z.B. über Triangulation bei WLAN-Headsets) und bei der Wiedergabe entsprechend berücksichtigt werden.

Combinatory Services

Da die Kommunikation im NGN über verschiedene Endgeräte in einem gemeinsamen Datennetz stattfindet, ist es möglich in einen Kommunikationsvorgang mehrere verschiedene Endgeräte gleichzeitig einzubeziehen. Über sogenannte *Combinatory Services* kann man beispielsweise Audio- und Videodatenströme für ein Gespräch mit verschiedenen Endgeräten empfangen und senden [dVel07, S.138].

Ein Beispiel für diesen Dienst ist ein Videotelefonat in welchem die Set-Top-Box für IPTV am Fernseher für die Videowiedergabe genutzt wird, während eine Netzwerk-Kamera das eigene Videobild aufnimmt (Abb. 4.13). Der Audiodatenstrom kann dabei beispielsweise über ein WLAN-Headset übertragen werden.

Für die getrennte Übertragung von Audio- und Videodaten wie bei Multichannel Media, dass eine Synchronisation erforderlich ist, da sonst Laufzeitunterschiede zwischen den Datenströmen Qualitätseinbußen nach sich ziehen. Eine Verbesserung der Übertragung erreicht man, indem die lokale NGN-Telefonanlage den Video- und den Audiodatenstrom zu einer SIP-Sitzung zusammenfasst und synchronisiert.

Um eine SIP-Verbindung auf mehreren Endgeräten nutzen zu können, wird ein Medienserver benötigt, welcher die eingehende SIP-Verbindung als UA terminiert, und zu jedem der beteiligten Endgeräte eine SIP-Verbindung aufbaut. Nach der Einleitung der Sitzung müssen die eingehenden Multimediadatenströme zu den entsprechenden

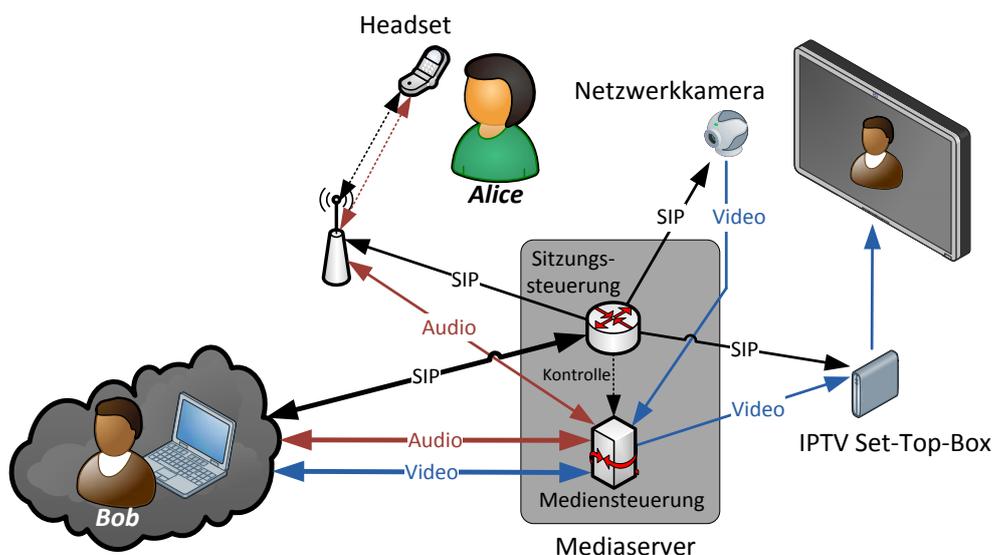


Abbildung 4.13: Combinatory Services ermöglichen die Nutzung einer NGN-Verbindung durch verschiedene Endgeräte. Dafür muss ein Mediaserver die SIP-Verbindung und den Multimediadatenstrom auf mehrere Endgeräte aufsplitten.

Endgeräten weitergeleitet und die ausgehenden Datenströme zu einer Sitzung zusammengefasst werden.

Community Cooperation

Die Kombination von Videokonferenz- und Präsenzdiensten ermöglicht eine entfernte Zusammenarbeit oder eine virtuelle Präsenz (*Community Cooperation*) [ETSI181003, S.62].

Damit können Personen an unterschiedlichen Standorten zusammenarbeiten und besser miteinander interagieren als über herkömmliche Kommunikationsverbindungen. Über Videokonferenzsysteme können die Nutzer emotionale Reaktionen ihrer Gesprächspartner sehen und so einen deutlich realistischeren Eindruck ihres Gegenüber gewinnen als bei einem Telefongespräch. Die Nutzung solcher Systeme ist nicht auf berufliche Kontakte beschränkt. Eine Verwendung in der Lehre oder Freizeit ist ebenso denkbar, da persönliche Beziehungen dort die Basis für jede Kommunikation sind. Mit einer entsprechenden Dienstgüte wäre so sogar eine virtuelle Probe einer Musikgruppe denkbar. Ein solcher Dienst würde eine „virtuelle Nähe“ über einen Kommunikationskanal ermöglichen, die innerhalb der heutigen Kommunikationsnetze nicht realisierbar ist. Dies könnte die beruflichen Beziehungen zwischen räumlich entfernten Kollegen oder etwa Fernbeziehungen deutlich beeinflussen.

Solche Konferenzsysteme sind heute bereits im Einsatz, beispielsweise Cisco TelePresence (Abb. 4.14). Sie werden aber aufgrund der hohen Anschaffungspreise meist nur genutzt, um Meetings mit langen Anreisewegen zu ersetzen [Hill07].



Abbildung 4.14: Videokonferenzsystem von Cisco, ermöglicht die gleichzeitige Wiedergabe von Videobildern der Gesprächspartner (Hintergrund) und die Anzeige einer Präsentation (Bildmitte) (Quelle⁶)

Ein Konferenzsystem besteht in der Regel aus einem Endgerät, welches als SIP UA die Video- und Audiodaten verarbeitet und mehrere Ein- und Ausgabegeräte ansteuert. Als Ausgabegeräte können Flachbildschirme (für Videobilder der Gesprächspartner), Beamer oder Monitore (für Präsentationen) und Lautsprecher (für Audiodaten) zum Einsatz kommen. Zusätzlich kann auch auf einen XMPP-basierenden Dokumentenserver (siehe Kapitel 7.4.1) zurückgegriffen werden, über welchen verteilt und in Echtzeit an kollaborativ gearbeitet werden kann. Professionelle Konferenzsysteme sind häufig so in Besprechungsräume integriert, dass mehrere Mikrofone fest im Tisch integriert sind und mehrere Kameras auf die Gesprächsteilnehmer gerichtet sind.

Die Steuerung des Videokonferenzsystems erfolgt häufig über ein anderes SIP-Endgerät (beispielsweise ein IP-Telefon), über welches die Verbindung zu dem zweiten Videokonferenzsystem eingeleitet wird. Die Funktionalität entspricht dabei weitestgehend der in Abb. 4.13 dargestellten *Combinatory Services*, mit dem Unterschied dass ein solches Konferenzsystem für den Nutzer auch nur eine SIP URI hat, und die weite-

⁶Cisco Telepresence <http://www.cisco.com/en/US/products/ps7060/index.html>

re Kommunikation mit den einzelnen Endgeräten nicht zwangsläufig über SIP erfolgen muss, da dieser Teil häufig herstellerspezifisch implementiert ist.

4.3.4 Alarmierungsdienste

Alarmierungsdienste senden dem Nutzer eine Benachrichtigung beim Eintritt eines bestimmten Ereignisses [BuTr07]. Dieses Ereignis kann je nach Anwendungsgebiet etwa durch Sensoren oder durch eine Zeitsteuerung eintreten.

Überwachung

Alarmierungsdienste können beispielsweise zur Überwachung von Maschinen eingesetzt werden. Der Nutzer erhält dann bei einer Störung der Maschine eine Nachricht. In herkömmlichen Netzen kann eine solche Mitteilung schon per SMS zugestellt werden, wobei mit 160 Zeichen der Inhalt stark begrenzt ist. Im NGN besteht aber die Möglichkeit deutlich mehr Daten zu übertragen, wobei hier Text, Bilder, Sprache oder sogar Video zum Einsatz kommen können um eine detaillierte Schilderung der Störung zu ermöglichen.

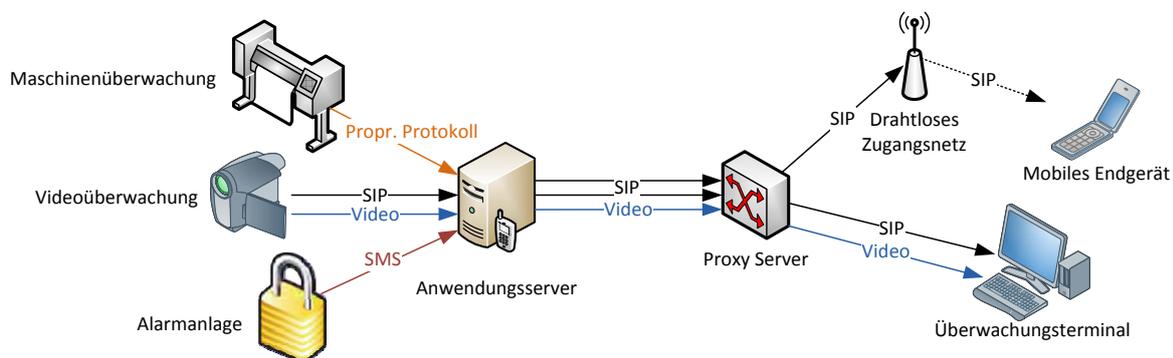


Abbildung 4.15: Für einen Alarmierungsdienst ist ein Anwendungsserver mit verschiedenen zu überwachenden Systemen verbunden, die auch über proprietäre Protokolle kommunizieren können. Im Falle einer Störung oder eines Ereignisses kontaktiert der Anwendungsserver die entsprechenden Nutzer.

Ein solcher Dienst könnte durch die Kombination mit Steuerungsdiensten (siehe Kapitel 4.3.7) zu einem komplexen Überwachungsdienst ausgebaut werden, bei dem der Nutzer nach der Benachrichtigung die Probleme je nach Ursache durch eine Remotesteuerung direkt beheben kann. Überwachungsdienste können natürlich auch in anderen Gebieten eingesetzt werden, wie die Überwachung von Sicherheitsbereichen. Dem Nutzer wird so etwa eine Benachrichtigung zugestellt wenn eine Tür unberechtigt

geöffnet wird, oder eine Überwachungskamera eine unberechtigte Bewegung detektiert hat.

Zur Realisierung des Überwachungsdienstes werden Sensoren oder Endsysteme benötigt, die bei den definierten Ereignissen den Anwendungsserver kontaktieren, damit dieser den Nutzer informieren kann. Diese Endsysteme können selber SIP UAs sein und den Anwendungsserver über das NGN kontaktieren. Sie können dabei beispielsweise im Falle einer intelligenten Überwachungskamera die verdächtigen Videodaten ebenfalls übermitteln, so dass der Nutzer nicht bloß per SIP-Message über das Problem informiert wird, sondern sich gleich ein Bild der Situation machen kann. Über den Anwendungsserver lassen sich außerdem mehrere verschiedene Überwachungsfunktionen, die auf verschiedenen Protokollen basieren, so zusammenfassen, dass das Endgerät des Nutzers nur einen einzigen Nachrichtentyp unterstützen muss. Falls die überwachten Endsysteme selbst die Funktionen eines SIP-UAs haben, können sie auch direkt einen Nutzer kontaktieren und so eine dezentrale Überwachung ermöglichen. Über einen Anwendungsserver lassen sich aber auch weitere Funktionen realisieren, etwa die Benachrichtigung einer ganzen Gruppe von Anwendern.

Zeitgesteuerte Alarmierung

Alarmierungsdienste lassen sich auch in Verbindung mit zeitgesteuerten Benachrichtigungen einsetzen. In diesen Bereich fällt beispielsweise der automatische Weckdienst, wie er von den Telefonanbietern schon lange eingesetzt wird.

Diese Anwendung lässt sich im NGN auf beliebige Ereignisse ausdehnen, da für die Benachrichtigung selbst nicht unbedingt ein Anruf erfolgen muss. Falls die Benachrichtigung durch eine Textnachricht erfolgt, wird der Benutzer nicht bei seiner aktuellen Tätigkeit unterbrochen und kann die Information später entgegennehmen.

So könnte der Anwendungsserver Zugriff auf einen öffentlichen Kalender (z.B. Veranstaltungskalender) haben, und dem Nutzer dann entsprechend seiner Vorlieben täglich Nachrichten mit passenden Veranstaltungsvorschlägen schicken. Dazu muss der Anwendungsserver eines der Kalenderformate (z.B. iCal) unterstützen und die relevanten Einträge in Timerereignisse umwandeln, die dann eine SIP-Nachricht an den Nutzer auslösen. Eine Anbindung an bestehende Groupwaresysteme (z.B. Microsoft Exchange) ist ebenfalls denkbar.

4.3.5 PIM-Dienste

Personal-Information-Management-Dienste⁷ (PIM-Dienste) bieten im NGN ein großes Potential, da eine Speicherung von Nutzerdaten, wie Kontakten und Terminen, an einem zentralen Punkt im Netz den Wechsel zwischen verschiedenen Endgeräten (z.B. Mobil und Stationär) für den Nutzer vereinfacht.

Contact List Overlay

Ein *Contact List Overlay* ist eine netzbasierende Kontaktliste welche regelmäßig durch das Endgerätes abgerufen wird. Änderungen an Daten am Endgeräte erfolgen direkt auf der Kontaktliste im Netz. Dies vereinfacht die Datenhaltung, besonders bei der Verwendung von mehreren Endgeräten für verschiedene Zugangsnetze [dVel07, S.139] (vgl. Device Unify Service, Kapitel 4.3.1), verhindert aber auch einen Datenverlust beim Verlust des Endgerätes.

Für die Speicherung der Daten wird ein Anwendungsserver mit einem Verzeichnisdienst wie LDAP benötigt. Das Endgerät kann dann jederzeit per LDAP auf die Kontaktliste zugreifen, die zur Nutzererkennung des Anwenders gehört. Es können aber auch netzwerkbasierende Synchronisationsprotokolle für mobile Endgeräte genutzt werden, wie ActiveSync, um sich in regelmäßigen Abständen mit einem zentralen Groupware-server zu synchronisieren.

Terminplaner

Ein *Terminplaner* kann innerhalb von einem PIM-Dienst auch mit weiteren Funktionen, wie Rückruf oder Benachrichtigung per IM oder MMS als Erinnerung und Memofunktion mit Speech-to-Text ausgestattet werden [BuTr07] (vergleiche Kapitel 4.3.4). Damit erhält man ein Werkzeug, um mit unterschiedlichen Endgeräten auf den Terminplaner zuzugreifen. Besonders die Sprachsteuerung ist wichtig, damit man auch mit herkömmlichen Legacy-Telefonen den Dienst nutzen kann.

Denkbar wäre auch ein Adapter zu Microsoft Exchange oder Google Mail, um bestehende Groupware-Systeme einzubinden. PIM-Dienste lassen sich auch nicht vollständig von Nachrichtensystemen trennen. So liegt ein *E-Mail-Vorlesedienst* eigentlich schon auf der Hand [BuTr07] und kann auch direkt zu einem *Voicemail-Dienst* erweitert werden [LeTO07a].

⁷Personal Information Manager (PIM) sind Programme zur Verwaltung von persönlichen Daten wie Kontakten, Terminen, Aufgaben, Notizen usw., wobei teilweise auch die Verwaltung von Dokumenten wie Mails und Briefverkehr dazugezählt wird. [Dysz07, S.42],[JoTe07]

4.3.6 Informationsdienste

Informationsdienste sollen einen einfachen Zugang zu Daten- und Nachrichtenquellen über NGN-Endgeräte ermöglichen. Diese Informationen können automatisch auf das Endgerät des Nutzers übertragen werden oder werden dem Nutzer nach entsprechenden Anfragen übermittelt.

Nachrichten- und Aktiendienste

Ein einfacher Informationsdienst ist die Abfrage von aktuellen Nachrichten oder Aktienkursen Sprachsteuerung [BuTr07]. Im NGN ist prinzipiell auch eine Abfrage oder Übermittlung der Informationen über andere Nachrichtentypen möglich, etwa per MMS oder als SIP-Nachrichten. Da viele Endgeräte aber auch über integrierte Webbrowser verfügen, ist die Verwendung von speziell an mobile Endgeräte angepassten Webseiten hier naheliegender als die Nutzung spezieller SIP-Nachrichten.

Voice-RSS

Eine Möglichkeit bestehende Abbonnementsysteme für Nachrichten einzubinden ist die Verwendung bestehender RSS-Feeds (Really Simple Syndication). Diese Feeds werden häufig als Nachrichtenticker eingesetzt und enthalten die Überschrift einer Webseite, einen kurzen Textabriss und einen Link zur ursprünglichen Seite.

Der Abruf dieser Feeds kann einerseits über *Voice-RSS* [jNet09] erfolgen, so dass der Nutzer die Inhalte der RSS-Feeds via Text-to-Speech über eine Sprachverbindung nutzen kann. Andererseits können je nach Fähigkeit des Endgerätes können die Inhalte auch als Textnachrichten übermittelt werden.

RSS-Feeds können auf beliebigen Internetseiten verwendet werden um über aktuelle Inhalte oder Änderungen zu informieren. Mit einem Adapter zu RSS-Feeds wäre beispielsweise auch eine Nutzung des beliebten Mikroblogs *Twitter*⁸ möglich.

Abfragesysteme

In [LeTO07a] wird vorgeschlagen die Signalisierung des SIP-Protokolls für ein Abfragesystem zu nutzen. So kann der Nutzer über spezielle SIP-URIs Informationen vom Anwendungsserver erhalten, welcher diese wiederum aus beliebigen Datenbanken bezieht.

⁸<http://www.twitter.com>

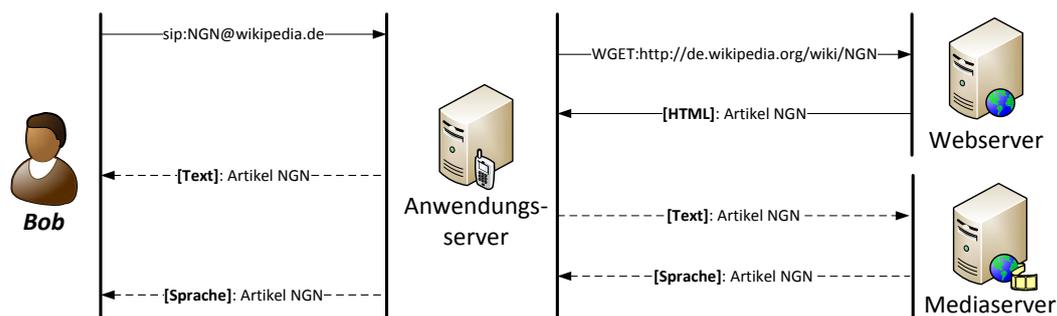


Abbildung 4.16: Dienst zur Abfrage eines Onlinelexikons via SIP-URI. Bei Bedarf kann ein Mediaserver in die Kommunikation einbezogen werden, um die Informationen als Sprachnachricht zu übertragen.

So kann der Nutzer die URI `sip:NGN@wikipedia.de` an den Anwendungsserver senden, welcher daraufhin den Artikel *NGN* aus der deutschsprachigen Wikipedia an den Anwender zurückschickt. Der Inhalt des Artikels kann als Text oder durch die Verwendung eines Mediaservers mit Text-to-Speech-Funktionalität (TTS) als Sprachnachricht an den Nutzer übertragen werden.

Digitale Bibliothek

Die vorgestellten Abfragesysteme lassen sich auch auf eine *Digitale Bibliothek* erweitern [ETSI181003, S.68]. Dabei kann der Nutzer über ein Zugangssystem entsprechende Literatur auswählen und sich anschließend auf ein passendes Endgerät übertragen lassen. Dabei kann das Endgerät zur Auswahl der Literatur durchaus ein normales NGN-Endgeräte wie ein Mobiltelefon sein, und die Literatur anschließend auf einen separaten E-Book-Reader übertragen werden. Die Bibliothek kann ebenfalls mit einer TTS-Funktionalität ausgerüstet werden, womit man eine Vorlesefunktion für die Titel der Bibliothek erhält.

4.3.7 (Haushalts-) Steuerungsdienste

Dienste zur entfernten Steuerung und Automation von Geräten sind ein relativ weitläufiges Feld, haben aber in Privathaushalten bisher kaum Anwendung gefunden. Generell sind für solche Steuerdienste aber auch andere Anwendungsgebiete, wie die Fernsteuerung von Industrieanlagen, denkbar.

Mediacontroller

Eine *SMS-Jukebox* [jNet09, Stee07] ist eine Anwendung, bei welcher beispielsweise ein Kunde im Restaurant via SMS die Musikauswahl im Lokal über sein Handy beeinflussen kann.

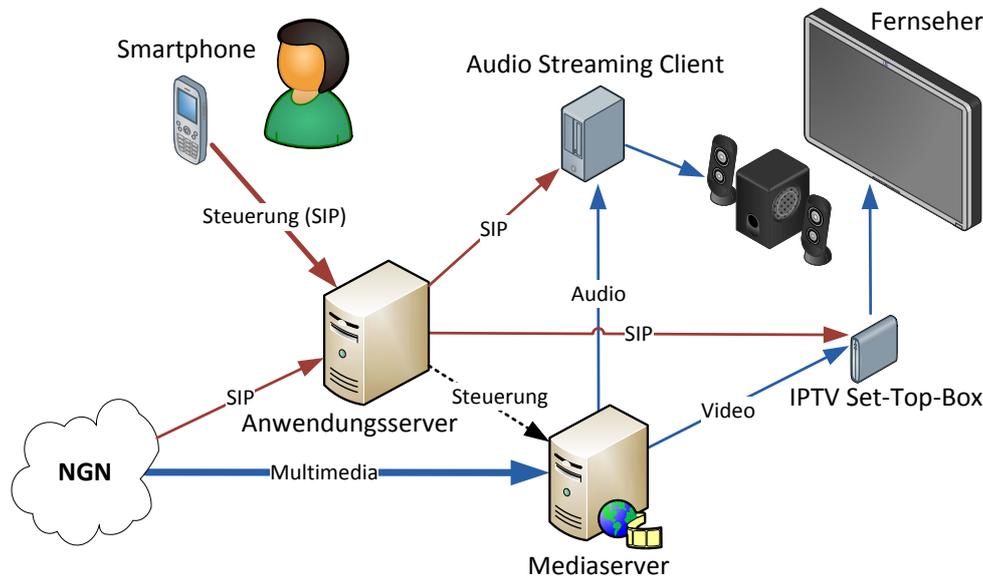


Abbildung 4.17: Über einen Anwendungsserver der eine Mediacontroller-Funktionalität bereitstellt, kann mit einem beliebigen NGN-Endgerät andere Geräte fernbedient werden. Kommt ein Mediaserver zum Einsatz, kann jeder Streamingclient, unabhängig von seiner eigenen Funktionalität, gesteuert werden.

Dieses Konzept lässt sich über ein NGN-Endgerät zu einem *Mediencontroller* aufwerten, der aus einem einfachen SIP-Telefon eine Universalfernbedienung macht [dVel07, S.135] (Abb. 4.17). Eine Nutzung dieser Idee ist erst durch die veränderte Abrechnung von Verbindungen im NGN und die Möglichkeit Steuersignale an beliebige Netzteilnehmer verbindungslos zu übertragen, möglich.

Außerdem können nun statt DTMF verschiedene Nachrichtentypen zur Übermittlung der Steuerungsdaten genutzt werden. Es können einerseits Nachrichtentypen von SIP genutzt werden, wie SIP-INFO oder SIP-NOTIFY, die Steuerdaten können aber auch über einen RTP-Datenstrom übertragen werden. Über einen Rückkanal können zusätzlich Menüinformationen in Form von Sprache, Text oder als Videostream übertragen werden. Für die Fernsteuerung einer Jukebox könnten so Informationen zum aktuell wiedergegebenen Titel auf dem Endgerät angezeigt werden.

Zur Realisierung können hier Endgeräte eingesetzt werden, welche die Steuerdaten

direkt als Sitzung selbst entgegennehmen, beispielsweise ein Streaming-Client der auch SIP-Nachrichten verarbeiten kann. Es kann aber auch ein Mediaserver ferngesteuert werden, wobei die Streaming-Clients völlig passiv sind. Der Mediaserver verarbeitet dann selbst die Steuerbefehle, die er vom Anwendungsserver übermittelt bekommt, und stoppt etwa einen Videostream, falls von der „NGN-Fernsteuerung“ ein entsprechender Befehl empfangen wird.

Hausautomation und Überwachung

Im Projekt *EURO-HOME* [ETSI181003, S.73] wird eine intelligente und flexible häusliche Umgebung vorgeschlagen. Dabei ist keine Einzelanwendung vorgesehen, sondern ein Dienst der prinzipiell unabhängig von den einzelnen verwendeten Komponenten funktioniert. Diese kommunizieren über ein hausinternes Datennetz mit dem Anwendungsserver, welcher die eigentliche Hausautomation zentral steuert.

Diese Steuerung kann verschiedene Bereiche umfassen, wie Beleuchtung, Heizung, Unterhaltungselektronik, Tür- und Schließsysteme oder Kommunikation. Die grundlegende Steuerung erfolgt anhand persönlicher Parameter, die der Nutzer aber nicht selbstständig konfigurieren muss. Verlassen etwa aller Personen das Haus, verschließt es selbstständig alle Türen und Fenster und aktiviert die Alarmanlage. Nähert sich der Nutzer wieder dem Haus, wird er über sein mobiles Endgerät aufgefordert ein Passwort einzugeben, um sich Zutritt zum Haus zu verschaffen.

Die Steuerung selbst kann außerdem zur Kostensenkung beitragen (wird das Haus verlassen, wird auch das Licht deaktiviert und die Heizung gedrosselt) und die Kommunikation unterstützen (befindet sich der Nutzer außerhalb des Hauses, werden alle Gespräche auf sein mobiles Endgeräte umgeleitet).

4.3.8 Location-based Services (LBS)

Durch die Nutzung von Informationen über den Standort des Nutzers, etwa über die Mobilfunkzellen in Reichweite oder ein integriertes GPS lassen sich die Dienste besser für Nutzer anpassen. Derartige Dienste sind teilweise schon realisiert, bieten aber noch viel Spielraum für neue Anwendungen. Im NGN ist bisher noch kein standardisiertes Datenformat zur Abfrage einer Nutzerposition festgelegt, so dass die meisten Anwendungen heute direkt auf dem Endgerät ausgeführt werden und dort direkt auf einen GPS-Empfänger zugreifen, um den Nutzerstandort zu bestimmen.

Ortsbezogene Verkehrsinformationen

Heute existiert bereits die Möglichkeit, dass Navigationssysteme in Fahrzeugen aktuelle Verkehrsinformationen per UKW-Radiosender (TMC) oder per Internet empfangen. Im TMC werden unabhängig von der Position alle Verkehrsinformationen aus dem Sendegebiet des TMC-Senders übertragen und haben eine relativ hohe Verzögerung (redaktionelle Bearbeitung). Bei der Nutzung von Daten aus dem Internet greifen die Hersteller der Navigationslösungen meistens auf proprietäre Verbindungen zurück. Teilweise werden Verkehrsbehinderungen aber auch über Bewegungsprofile und Durchschnittsgeschwindigkeiten auf bekannten Strecken ermittelt, die von den Nutzern an den Anbieter übermittelt werden.

Das NGN bietet hier die Möglichkeit, dass der Nutzer anhand seiner Position für ihn passende Informationen anfordern kann [ETSI181003]. Für den Datentransport kann dabei auf SIP-Nachrichten zurückgegriffen werden.

Ortsbezogene Suche

Ein weiterer Dienst ist eine ortsbezogene Branchensuche [LeTO07a, BuTr07]. Der Nutzer kann sich etwa ein italienisches Restaurant in seiner Umgebung suchen lassen und wird anschließend auf Wunsch zur Reservierung eines Tisches mit dem Restaurant verbunden. Je nach Genauigkeit der Ortung ist anschließend auch eine Wegbeschreibung zum Ziel möglich.

Die Position des Nutzers kann auch an den Gesprächspartner übermittelt werden, so dass der Nutzer beim Anruf in der Taxizentrale beispielsweise nicht mehr sagen muss wo er sich befindet (unabhängig davon, ob er seine Position überhaupt kennt). Diese Technik wird heute schon teilweise in Notrufzentralen eingesetzt, wenn ein Anrufer der Ortung seines Mobiltelefones zustimmt oder wenn er sich in einem Fahrzeug mit GPS und Notrufsystem befindet.

Eine besondere Verknüpfung einer kommerziellen Suche mit bezahlten Suchergebnissen und eines Online-Navigationssystems bietet Google seit kurzem in den USA an [Goog09]. Dabei kann der Nutzer neben den Navigationsfunktionen seines Mobiltelefons gleichzeitig eine semantische sprachgesteuerte Suche über Ziele in der Umgebung nutzen (Beispiel: „Suche das Museum mit dem babylonischen Ishtar-Tor“, was den Nutzer zum Pergamonmuseum in Berlin führt). Dieser Dienst erfordert jedoch wieder eine Anwendung, die auf dem Endgerät des Nutzers ausgeführt wird.

Meet-Me-Service

Der in [ETSI181003, S.62] vorgeschlagene Dienst *Meet Me: Meet the Right Person at the Right Time* verbindet die Position des Nutzers mit Kontextinformationen über den Nutzer. Das Ziel ist passende Gesprächspartner in der Umgebung zu finden. So würde das System auf einer Messe Nutzer mit ähnlichen Interessen zusammenführen, ohne dass diese durch langwierige Gespräche nach interessierten Personen suchen.

Der Schwerpunkt bei diesem Dienst liegt auf der Erfassung und Auswertung der kontextsensitiven Informationen über den Nutzer. Dazu wird eine geeignete Datenbankstruktur zur Speicherung und ein Prozess zur Sortierung und Abfrage der Daten benötigt [Lewa07]. Für die Lokalisierung der Nutzer wird ebenfalls ein standardisiertes Protokoll benötigt, wobei hier nicht unbedingt die globale Position des Nutzers übertragen werden muss, sondern die räumliche Nähe zu anderen Nutzern (etwa durch Bluetooth-Geräte in Reichweite) als Standortmerkmal für die Suche ebenfalls ausreichend ist. Der Anwendungsserver wertet anschließend die Nutzerstandorte aus und

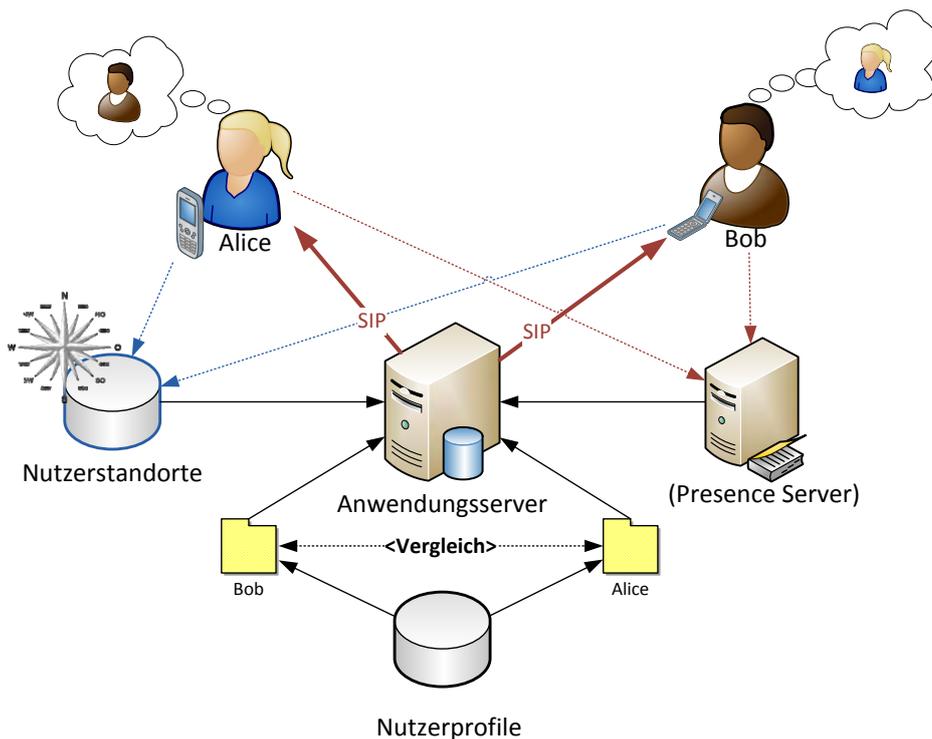


Abbildung 4.18: Der Meet-Me-Service sucht Nutzer mit ähnlichen Interessen die sich in der Nähe befinden und stellt anschließend einen Kontakt her.

sucht in den Nutzerprofilen nach Übereinstimmungen. Findet er eine passende Paarung, benachrichtigt er beide Nutzer, so dass sie die Möglichkeit haben außerhalb des Kommunikationsnetzes Kontakt aufzunehmen. Der Dienst kann über einen Presence

Server so erweitert werden, dass der Anwendungsserver den aktuellen Kommunikationszustand der Nutzer mit in die Bewertung einbezieht, so dass nur Nutzer kontaktiert werden, die sich gerade nicht in einem Gespräch befinden.

Do-not-Disturb-Mode

Mobile Endgeräte können aufgrund ihrer Position, beispielsweise über den genutzten Access-Point im Beratungsraum, oder benachbarte Geräte (Dienstberatung) selbstständig in einen *Do-not-Disturb-Mode* wechseln. Sie erkennen so, dass der Nutzer sich gerade in einer Situation befindet, in welcher er nicht durch Anrufe unterbrochen werden möchte.

Zur Umsetzung bietet sich hier der Präsenzdienst an, bei welchem der Anrufer den Status des Nutzers abfragen kann. Je nach Konfiguration des Dienstes kann der Anrufer auch eine entsprechende Meldung erhalten und sein Gespräch wird abgewiesen.

On-Line-Guide

Ein weiterer ortsbezogener Dienst, der Informationen abhängig der Nutzerposition liefert, ist ein *On-Line Guide* für Touristen [ETSI181003, S.17]. Dieser liefert dem Nutzer Informationen über Sehenswürdigkeiten in der Umgebung. Der Dienst kann aber auch genutzt werden, um eine virtuelle Führung durch ein Areal bereitzustellen. Dem Nutzer werden dann an den entsprechenden Stationen Informationen zu den sehenswerten Objekten vorgelesen, vergleichbar mit einem Audioguide.

Als Datenquelle hierfür kann beispielsweise das freie Online-Lexikon Wikipedia verwendet werden, das für viele Einträge sogenannte Geo-Tags mit der genauen Position des beschriebenen Objektes enthält.

Ein Projekt, das eine ähnliche Funktionalität als Ziel hatte, ist das „*Touristisches Assistenzsystem für barrierefreie Urlaubs-, Freizeit- und Bildungsaktivitäten*“ (TAS). Dort lag der Fokus auf der touristischen Erschließung für körperlich behinderte Menschen, indem auf Basis ihrer körperlichen Befähigung und ihrer Interessen touristische Routen und Ziele in ihrer Umgebung bereitgestellt werden [LuWi07].

4.4 Bewertung und Ausblick

Die beschriebenen Dienste stellen lediglich einen Ausschnitt über mögliche Dienste im NGN dar. Einige der Dienste sind bereits für Endnutzer verfügbar, auch wenn dabei

teilweise noch andere Protokolle (z.B. HTTP bei Webservices) zum Einsatz kommen oder der Dienst als Anwendung auf mobilen Endgeräten ausgeführt wird.

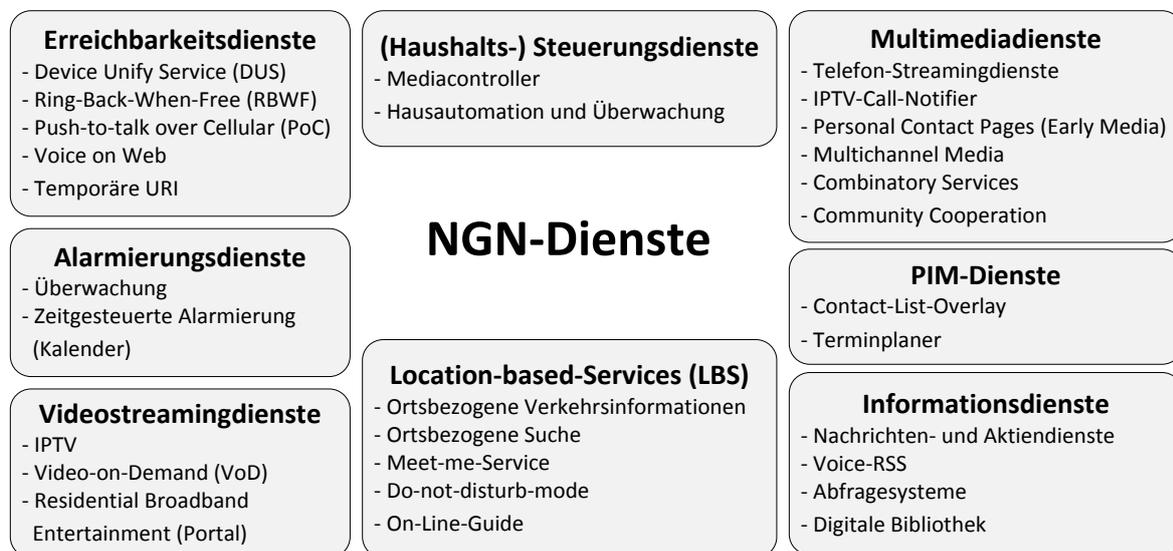


Abbildung 4.19: Übersicht über die untersuchten Dienstklassen und Dienstkonzepte

Die Einführung von leistungsfähigeren Spracherkennungs- und Synthesystemen werden Dienste, die mit natürlicher Sprache arbeiten, in den nächsten Jahren deutlich verbessern. Besonders für die Nutzung der Dienste über das herkömmliche Telefonnetz ist diese Art der Interaktion wichtig. Durch bessere Spracherkennung und Synthese sind auch Dienste denkbar, die mit heutigen Techniken nur sehr eingeschränkt funktionieren würden, etwa eine Speech-to-Speech-Translation [BuTr07]. Damit könnte ein Anwendungsserver als Proxy zwischen zwei Gesprächspartnern, welche unterschiedliche Sprachen sprechen, automatisch in Echtzeit den Audiodatenstrom beider Teilnehmer in die jeweilige andere Sprache übersetzen.

Viel Spielraum für Kreativität besteht auch in Zusammenhang mit komplexen Diensten. Diese basieren auf einer Zusammenfassung und Erweiterung von anderen Diensten in neuen Diensten, die daraus einen größeren oder komplett neuen Funktionsumfang generieren. Da heute nur ein Bruchteil der gesamt möglichen Dienste im NGN bereits bekannt sind, lässt sich nur schwer beurteilen welche Möglichkeiten diese Dienste in den kommenden Jahren bieten werden.

Zur Erzeugung solcher Mehrwertdienste wurden bei der Spezifikation des NGN bereits verschiedene Schnittstellen zur Anwendungsschicht im Netz festgeschrieben. Diese Schnittstellen werden in eigenen Netzelementen, den Anwendungsservern realisiert. Diese Anwendungsserver werden mit ihren Schnittstellen und den zugrundeliegenden Plattformen im folgenden Kapitel untersucht.

5 Anwendungsserver für NGNs

5.1 Anwendungsserver als Teil des Netzes

5.1.1 SDP - Service Delivery Platform

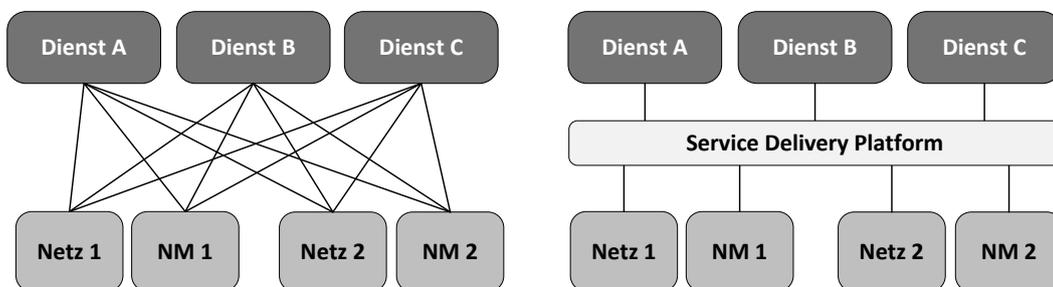


Abbildung 5.1: Service Delivery Platform (SDP) als gemeinsame Schnittstelle zum Netz und Netzmanagement (NM) für verschiedene Dienste in mehreren Netzen.

Zur Erbringung einer Dienstleistung muss eine Anwendungslogik, welche den Dienst ausführt, im Kommunikationsnetz Schnittstellen zum Transportnetz, zur Signalisierung und zu Managementeinrichtungen des Netzes haben. Typischerweise wird ein Dienst aber nicht nur für das Netz eines Providers oder für ein Zugangnetz implementiert, sondern soll in allen Netzen verfügbar sein und müsste so für jedes Netz separate Schnittstellen bereitstellen. Eine *Service Delivery Platform* (SDP) stellt diese Schnittstellen als eine gemeinsame Plattform für verschiedene Dienste bereit.

Im NGN sind die einzelnen Zugangnetze bereits zusammengefasst, wodurch ein Dienst, der ein Interface für das NGN besitzt, bereits über alle Zugangnetze kommunizieren kann. Die SDP kann aber trotzdem als Schnittstelle zwischen Diensten und dem NGN zum Einsatz kommen und implementiert dann einen definierten Satz von Schnittstellen, beispielsweise Parlay (siehe Kapitel 5.2.2.1) und stellt damit Zugänge zu Netzressourcen wie Protokollzugänge, Signalisierung, Abrechnung, Nutzerauthentifizierung etc. bereit. Die Service Delivery Platform wird im NGN durch das Netzelement des *Anwendungsservers* (auch *Application Server*) umgesetzt.

5.1.2 Definition Anwendungsserver

Ein *Anwendungsserver* (AS) ist ein System, welches verschiedene Dienste für die Nutzer im Next Generation Network zur Verfügung stellt. Die Inhalte und Funktionen, die ein solcher Dienst bereitstellen kann, wurden bereits in Kapitel 4.2.2 besprochen.

Ein SIP Application Server (SIP AS) ist eine Kombination aus SIP User Agent und/oder SIP Proxy Server und/oder SIP Redirect Server sowie einer Software-Plattform für die Dienste. [BuTr07, S. 109]

Eine zentraler Bestandteil eines AS ist die Programmierschnittstelle (API) über welche der Dienst implementiert werden kann. Die API stellt die Schnittstelle zwischen der Plattform des AS und den Kommunikationsfunktionen im Netz dar und ermöglicht die Entwicklung der Dienste in einer bestimmten Programmiersprache. Verschiedene Konzepte, die eine solche Schnittstelle erzeugen werden in Kapitel 5.2 betrachtet.

Zur Erzeugung von Diensten muss der AS verschiedene Protokolle, wie SIP, RTP, HTTP oder Diameter unterstützen. Für den Provider sind Schnittstellen zu Systemen, welche die Zugriffsteuerung, Nutzerauthentifikation und zur Abrechnung der Dienste ermöglichen, nötig. Zum Zugang solcher AAA-Systeme (Authentication-, Authorization-, Accounting-Systeme) kann beispielsweise das Diameter-Protokoll verwendet werden. Für den Versand und den Abruf von Mediendaten kann der AS auch andere Media-server steuern und die Verbindungsparameter über das Session Description Protocol (SDP) anpassen. Eine Schnittstelle zu Datenbanken kann für den Abruf von Inhalten (Informationsdienste, Alarmierungsdienste) oder für die Ermittlung von Gesprächspartnern/ Zielrufnummern (Erreichbarkeitsdienste, LBS) verwendet werden.

Prinzipiell können die Dienste auch von beliebigen Servern im Internet angeboten werden, die SIP-Verbindungen ermöglichen. Dabei die Verbindung zum Server dann natürlich nur noch ohne Dienstgüteverbesserung QoS nach dem Best-Effort-Prinzip genutzt werden.

Eine interessante Möglichkeit zur Erzeugung von komplexen Diensten ist der Aufruf von anderen Anwendungsservern. So kann man durch die Aggregation von mehreren Diensten einen neuen komplexen Dienst bereitstellen.

5.1.3 Anwendungsserver als Netzelement

Gegenüber des IN/CAMEL-Konzeptes von ISDN (siehe Kapitel 2.2) unterscheidet sich das Konzept der Diensterzeugung im NGN stark, da die Dienste auf einer eigenen Ebene ausgeführt werden und so weniger Verflechtungen mit dem Nutzdatentransport und

der Sitzungssteuerung haben. Im IN waren die Dienste in den Knoten des Netzes implementiert oder mussten dort zumindest bekannt sein. Im NGN sind die Dienste vollständig von den Netzknoten getrennt. Änderungen an der Plattform sind so einfacher möglich, da kaum noch Abhängigkeiten mit dem Kernnetz bestehen. Außerdem ist eine deutlich schnellere und günstigere Einführung der Dienste möglich, da teilweise auf bereits bestehende Dienste (z.B. Webservices) und Plattformen (z.B. Java Enterprise Environment) zurückgegriffen werden kann (siehe Kapitel 5.3).

Durch die Auslagerung der Dienste aus dem Kernnetz entsteht aber auch eine neue Konkurrenz durch Drittanbieter, denn im IN ist das Anbieten Dienstleistungen ein sogenannter *Walled Garden* [Berg06], also auf die Netzprovider beschränkt.

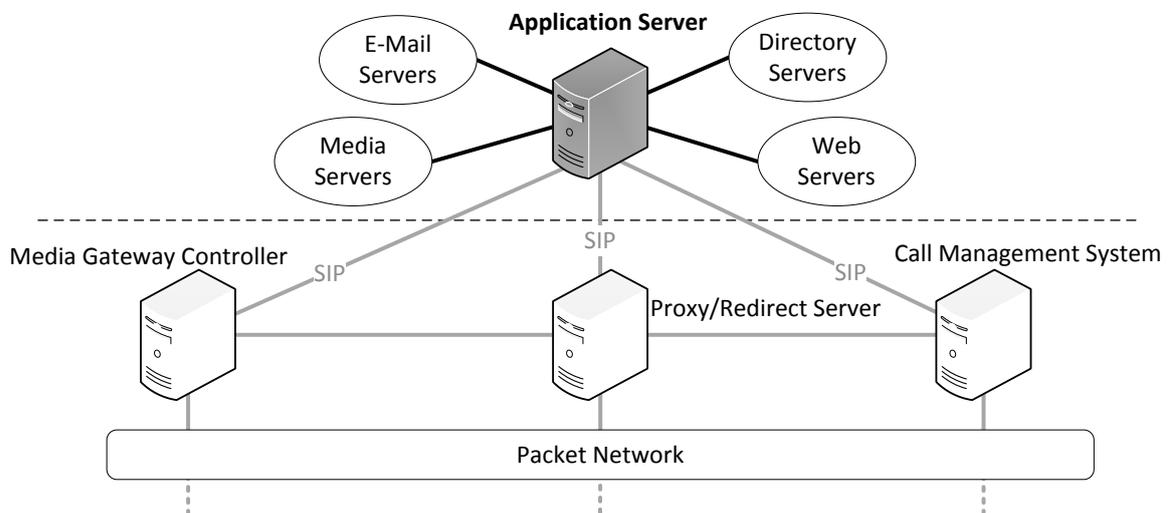


Abbildung 5.2: Der Anwendungsserver ist im NGN ein unabhängiges Netzelement, da er über verschiedenen Knotenpunkte im Netz kommunizieren und Endsysteme auch direkt ansprechen kann.

5.1.4 Betriebsmodi eines Anwendungsservers

Der Anwendungsserver kann als aktives Netzelement im NGN selbstständig SIP- und RTP-Verbindungen steuern, d.h. Verbindungen initiieren, terminieren oder vermitteln [LeTO07a]. Er agiert in der Regel als User Agent (UA), kann aber auch, wie bereits in Kapitel 4.2.3 besprochen, andere Rollen einnehmen.

Es existieren verschiedene Wege die Nutzung eines Dienstes einzuleiten. Danach können die Betriebsmodi eines Anwendungsservers unterteilt werden [LeTO07a, TrWe07]. Im einfachsten Fall etabliert der Nutzer eine Verbindung zum AS, wobei der AS dann als zweiter Nutzer (UA) das Gespräch annimmt. Über die Verbindung kann der Nutzer

anschließend Daten oder Inhalte (z.B. Videos, Audiomittschnitte) abrufen. Ein einfacher Dienst nutzt in der Regel nur einen Betriebsmodus, komplexere Dienste mit vielen Funktionen können aber auch mehrere Betriebsmodi für die Teilfunktionen nutzen.

Content-Delivery

Beim Betriebsmodus *Content-Delivery* löst der Nutzer den Dienst aus, indem er beispielsweise eine SIP-Nachricht an den AS sendet oder eine komplette SIP-Sitzung mit dem AS aufbaut. Der Anwendungsserver liefert anschließend entsprechend der Anfrage des Nutzers die gewünschten Inhalte.

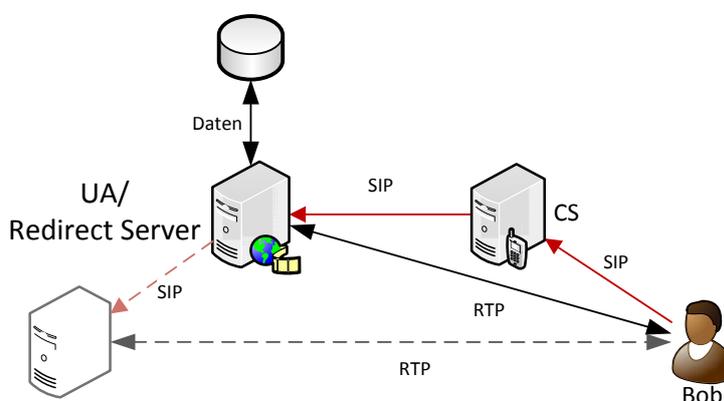


Abbildung 5.3: Betriebsmodus Content-Delivery, der Nutzer ruft über den Anwendungsserver Informationen von einer externen Datenquelle ab.

Terminiert der AS die Sitzung selbst, dann funktioniert er dabei als UA (siehe Kapitel 3.3). Es bestehen prinzipiell mehrere Wege Inhalte vom AS anzufordern. Die Anfrage kann beispielsweise direkt in der SIP-URI der INVITE-Nachricht (siehe Kap. 3.2.2.2) enthalten sein, so dass der Anwendungsserver beispielsweise für die URI `sip:NGN@wikipedia.de` den Inhalt des Artikels „NGN“ aus der deutschsprachigen Wikipedia zurücksendet [LeTO07a]. Das würde in diesem Fall jedoch nur dann funktionieren, wenn die Domain `wikipedia.de` etwas mit dieser Anfrage anfangen kann, oder sie entsprechend an einen passenden AS weiterleitet.

Wird beim Anwendungsserver ein Dienst angefordert, den er nicht selbst erzeugt, dann kann er als Redirect *Redirect Server* die Verbindung des Nutzers auf einen anderen Server umleiten und diesen Server die gewünschten Inhalte zum Nutzer ausliefern lassen. Für die Anforderung eines Dienstes beim AS stehen auch andere Möglichkeiten zur Verfügung, beispielsweise eine *MESSAGE-SIP*-Anfrage, welche innerhalb der Mitteilung einen speziellen Text enthält (etwa `FETCH FROM [wikipedia.de] ARTICLE [ngn] AS TEXT`), durch welchen der Dienst im AS ausgelöst wird. Besonders für die

Verwendung mit herkömmlichen Telefonen, die an das NGN angeschlossen sind, ist die Steuerung des Dienstes über Sprache und Tastentöne (DTMF) von Bedeutung. Dazu sollte der Anwendungsserver ein sprachbasierendes Menü verwenden (Text-to-Speech), damit der Nutzer auch über eine SIP-Sitzung mit einem Sprachkanal mit dem Dienst interagieren kann und entsprechende Inhalte abrufen kann.

Wake-Up

Einige Dienste erfordern einen Betriebsmodus, bei welchem der AS eine Anwendung selbstständig aufgrund vorgegebener Parameter startet. Ein solcher *Wake-Up*-Betrieb kann beispielsweise durch das Eintreten vorher festgelegter Ereignisse auf der Datenseite (Benachrichtigung einer Alarmanlage, Eintreffen eines Fußballergebnisses etc.) oder aufgrund eines Timers (Terminplaner, Wecker) ausgelöst werden. Bei diesem Betriebsmodus kontaktiert der AS den Nutzer selbstständig, indem er als *UA* eine SIP-Sitzung mit ihm aufbaut und anschließend eine Text- oder Sprachmitteilung mit dem Inhalt des Ereignisses überträgt.

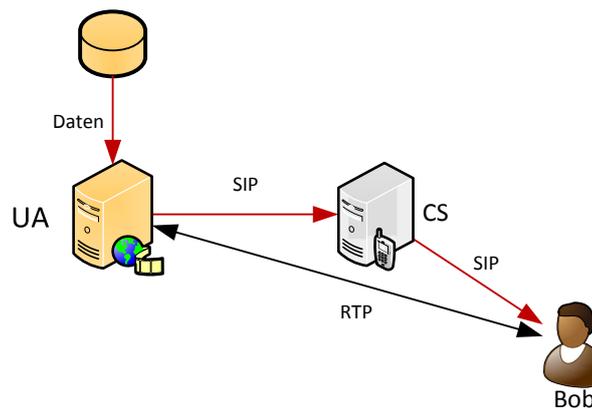


Abbildung 5.4: Betriebsmodus Wake-Up, durch ein externes Ereignis wird eine Sitzung zum Nutzer etabliert.

Call-Fowarding

Der Betriebsmodus *Call-Fowarding* ermöglicht dem Nutzer die Verwendung eines Dienstes, der ihn mit einem Gesprächspartner verbindet, den er nicht explizit ausgewählt hat. Dafür kontaktiert der Nutzer den AS mit einer bestimmten SIP-URI, der ein Dienst zugordnet ist (beispielsweise `sip:herrenfriseur@141.24.93.180`) und der AS wählt entsprechend vorher festgelegter Kriterien einen dazu passenden Teilnehmer (z.B. den Herrenfriseur der vom Nutzerstandort aus am schnellsten erreichbar

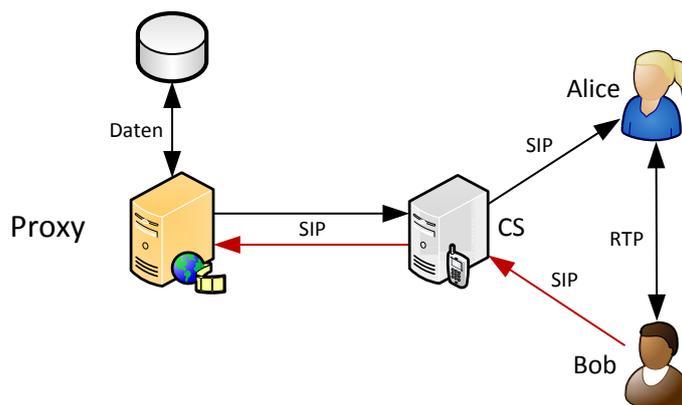


Abbildung 5.5: Betriebsmodus Call Forwarding, der Anwendungsserver sucht dabei nach verschiedenen Parametern ein geeignetes Rufziel für den Anrufer.

ist) und leitet das Gespräch zu diesem Teilnehmer weiter. Der AS funktioniert dabei als *Proxy Server* für die Verbindung.

Click-to-Dial

Ein AS kann auch als dritte Partei (3rd Party) Gespräche zwischen zwei Teilnehmern einleiten. Das ist ein typisches Szenario für *Click-to-Dial*. Der AS erhält beispielsweise durch den Nutzer über ein Webinterface den Befehl, eine Verbindung zwischen dem Nutzer und einem anderen Teilnehmer zu etablieren. Dafür leitet er als 3rd Party einen Anruf zwischen dem Endgerät des Nutzers und dem gewünschten Teilnehmer ein. Der AS ist in diesem Fall ein *Back-to-Back UA* (B2BUA).

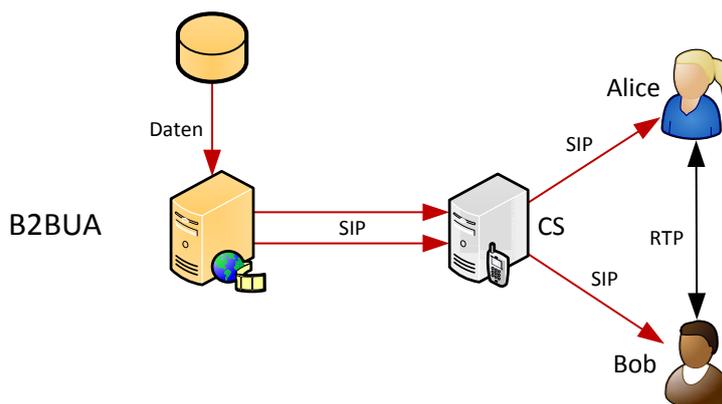


Abbildung 5.6: Betriebsmodus Click-to-dial, der Anwendungsserver initiiert durch ein Datenereignis als dritte Partei eine Sitzung zwischen zwei Nutzern.

Typischerweise leitet der B2BUA dabei nur die SIP-Sitzung ein. Der RTP-Datenaustausch erfolgt dann direkt zwischen den Teilnehmern. Der B2BUA kann aber auch die

RTP-Datenströme terminieren, falls keine passenden Audiocodex vorhanden sind, so dass er den Audiodatenstrom während der Sitzung für den jeweiligen Gesprächspartner in ein passendes Format umwandelt.

5.2 Technische Konzepte

5.2.1 High-Level vs. Low-Level-APIs

Für die Realisierung der Dienste im NGN wurden verschiedene Konzepte entwickelt. Diese lassen sich anhand ihrer Programmierschnittstellen (API) in zwei Gruppen unterteilen: *Low-Level-API* und *High-Level-API* [LeTO07b].

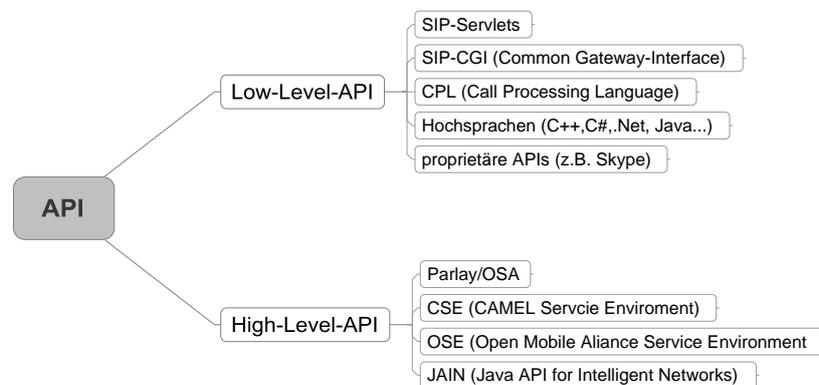


Abbildung 5.7: Übersicht über APIs für die Dienststeuerung im NGN

Low-Level-APIs sind Programmierschnittstellen, welche direkt auf Betriebssystemebene des Servers als Anwendung ausgeführt werden. Diese APIs implementieren alle Funktionen selbst, die im NGN benötigt werden (Kommunikation, Authentifizierung etc.) und sind typischerweise in Programmiersprachen wie C++, Java oder C# (.Net oder Mono) geschrieben. Diese APIs können auch proprietär sein, wie die Skype-API (siehe Kapitel 6.3.2).

Die High-Level-APIs dagegen basieren auf einer Middleware, welche als zusätzliche Softwareschicht oberhalb des OSI-Schichtenmodells arbeitet und für die Anwendungscontainer, die innerhalb der Middleware ausgeführt werden, den Datentransport, Funktionsaufrufe oder Transaktionssicherheit bereitstellt. Bekannte Anwendungen hierfür sind etwa der *WebSphere Application Server* von IBM oder die *SAP Exchange Infrastructure*.

Allgemein ist der Vorteil der High-Level-APIs der große Funktionsumfang, den sie für Anwendungen mitbringen. Gegenüber Low-Level-APIs ist dafür die Ausführungsge-

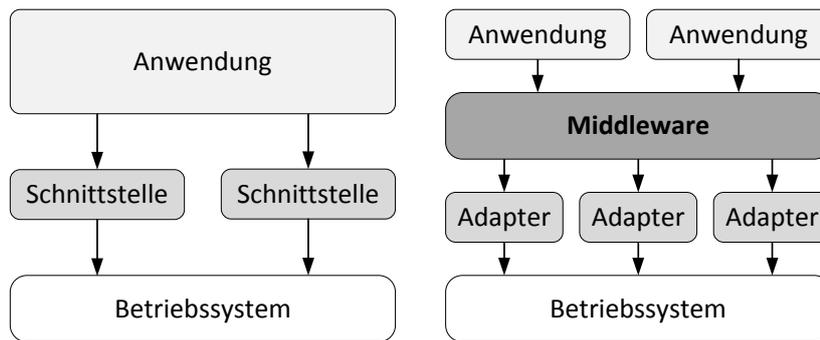


Abbildung 5.8: Low-Level-APIs müssen alle Schnittstellen zum Betriebssystem oder zu Protokollen selbst implementieren, wohingegen High-Level-APIs auf einer Middleware aufbauen, die diese Schnittstellen bereitstellt.

schwindigkeit deutlich geringer und strukturelle Anpassungen sind deutlich schwerer, da die Architektur der Anwendungen durch die Middleware bereits vorgegeben ist. Im Bereich der Schnittstellen zur Erzeugung von Diensten im NGN spricht man bei High-Level-APIs von sprachunabhängig definierten Schnittstellen, die nur über Abläufe und den Funktionsumfang definiert sind. Die Spezifikation der Schnittstelleneigenschaften sind meistens in UML verfasst. Die konkrete Implementierung von Schnittstellen zum NGN in einer bestimmten Programmiersprache sind Low-Level-APIs, da sie den Funktionsumfang als Betriebssystemaufrufe oder als Netzwerkkommunikation direkt umsetzen.

5.2.2 High-Level-Lösungen

5.2.2.1 Parlay/OSA

Das Ziel der *Parlay Group* ist die Erzeugung einer API für Anwendungen, die über ein unsicheres IP-Netzwerk Dienste im sicheren Kernnetz des Netzanbieters bereitstellen können. Diese API definiert verschiedene Schnittstellen, die Methoden, Events und Parameter besitzen, mit denen externe (untrusted) und interne Anwendungsanbieter den Zugriff und die Kontrolle über das Kernnetz (bzw. Teile davon) haben [BBDU00].

Ziel der API ist die Bereitstellung von Diensten, die auf der Software- und Hardwareinfrastruktur des Diensteanbieters laufen und gleichzeitig über eine unsichere Verbindung mit dem Kernnetz kommunizieren. Dazu gehört ein Parlay API Framework, welches den Zugriff auf die Parlay-Dienste zur Netzkontrolle steuert und verschlüsselt. Die Parlay-Dienste selbst ermöglichen die Kommunikation über SIP, H323 oder auch über nicht-NGN-Technologien wie ISDN.

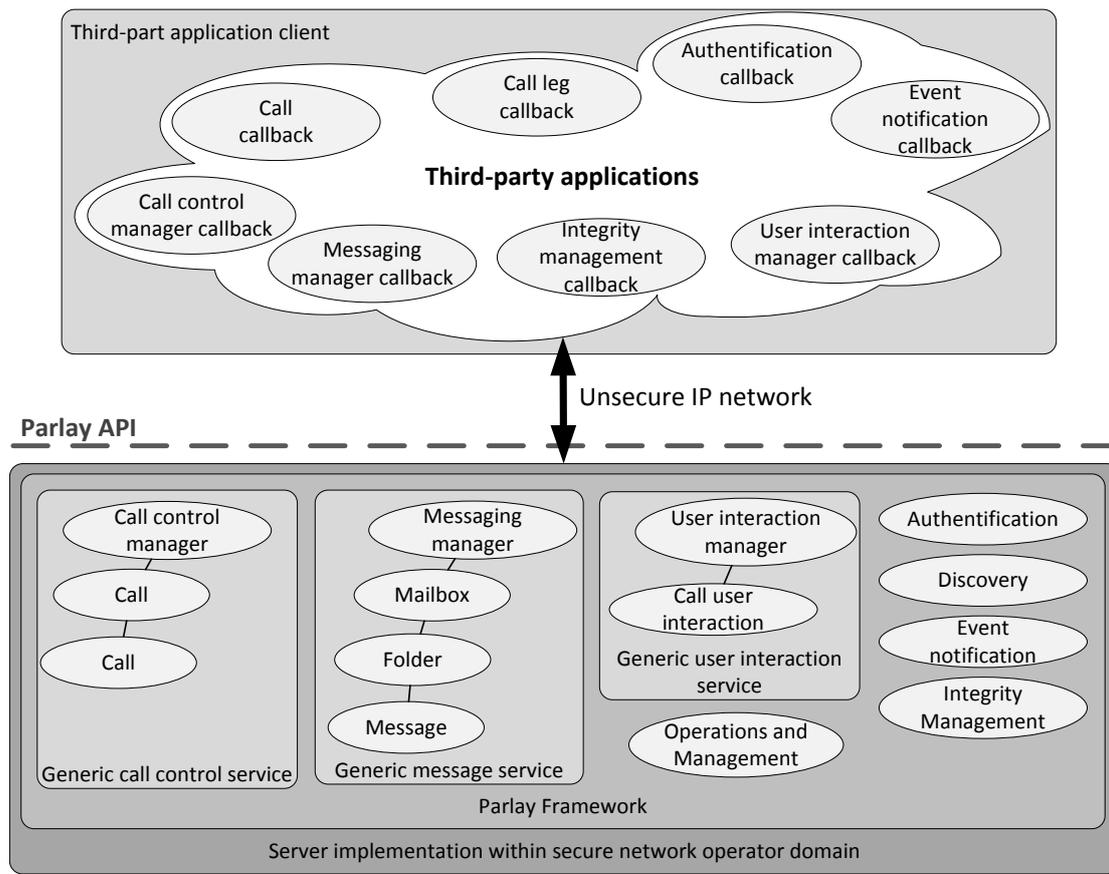


Abbildung 5.9: Architektur der Parlay 1.2 API für die Bereitstellung von Diensten im NGN über eine unsichere IP-Verbindung (Quelle: [BBDU00]).

Die bereitgestellte Schnittstelle wird als *Open Service Access (OSA)* bezeichnet und bietet die Erkennung, Registrierung, Authentisierung und eine Zugriffsteuerung für den „fremden“ AS. Diese Funktionalität bieten die Netzelemente (S-CSCF) von sich aus nicht [LeTO07b]

5.2.2.2 OMA Service Environment (OSE)

Die *OMA Service Environment (OSE)* der *Open Mobile Alliance (OMA)* stellt auf der Basis der Kerntechniken von IMS sogenannte Service-Enabler bereit, worauf Anwendungsserver dann zugreifen können. Die Enabler stellen Funktionalitäten, wie beispielsweise einen Präsenzdienst, bereit, die von anderen Anwendungen genutzt werden können. Der Aufbau und die Funktionsweise ähneln dem Parlay-Konzept sehr stark.

5.2.2.3 Camel Service Environment (CSE)

Für die Nutzung bereits bestehender Dienste aus dem Intelligenten Netz auf Basis von IN/CAMEL (siehe Kapitel 2.2) existiert für das NGN die *Camel Service Environment* (CSE). Diese Schnittstelle ist nicht direkt zur Entwicklung von neuen Diensten vorgesehen, sondern soll die bestehenden Dienste ohne große Anpassungen im NGN nutzbar machen [LeTO07a]. Dafür existieren Protokollumsetzer, die SIP in CAP (Camel Application Part) konvertieren.

5.2.2.4 Java APIs for Integrated Networks (JAIN)

Die *Java APIs for Integrated Networks* (JAIN) ist eine Spezifikation der Parlay API (Kap. 5.2.2.1) in Java, um eine weite Verbreitung von Parlay in der Industrie zu erreichen [BBDU00]. JAIN wurde durch die *JAIN Parlay Edit Group* Anfang 2004 veröffentlicht [Sun04a].

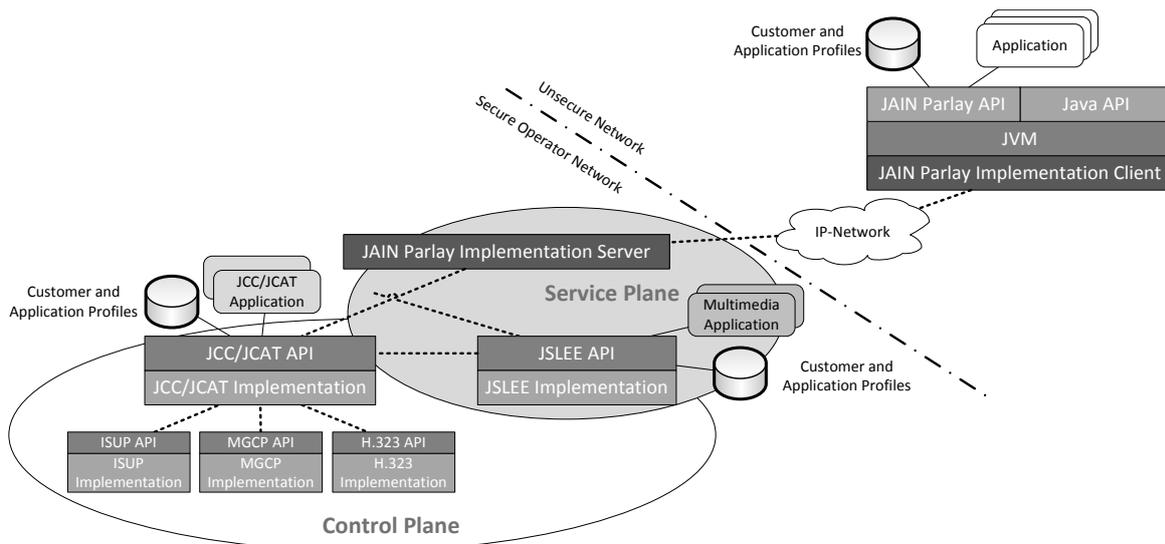


Abbildung 5.10: Modell der JAIN-Implementierung der Parlay-Spezifikation (frei nach [BBDU00])

Der Unterschied zur ursprünglichen Parlay-Spezifikation ist die fehlende Aufteilung zwischen Parlay-Server und dem Client zur Verbindung des Anwendungsservers mit dem Kernnetz. Die Spezifikation setzt die Parlay-Schnittstelle direkt im Anwendungsserver als API um, so dass die Übertragungstrecke providerspezifisch realisiert werden kann. Die JAIN-Spezifikation enthält verschiedene Komponenten zur Umsetzung von Diensten in SIP-basierenden Netzen [Sun04b]. Der Kern der Spezifikation ist die *JAIN Service Logic Execution Environment* (JSLEE), eine auf der *Java Enterprise Edition*

(JavaEE) basierende Plattform zur Implementierung von Diensten in Kommunikationsnetzen. Dazu gehören aber auch andere APIs wie das *JAIN Session Initiation Protocol* (JSIP) oder die *SIP API for J2ME* (SIP for J2ME).

Aufgrund der Implementierung der Parlay-Schnittstellen auf Basis der existierenden Middleware Java Enterprise Environment (JavaEE) ist diese API eine High-Level-API. In einigen Veröffentlichungen (z.B. [LeTO07b]) wird sie jedoch auch als Low-Level-API bezeichnet, da die JavaEE-Plattform, welche alle Funktionen für die Kommunikation bereitstellt, durch JSIP direkt um die SIP-Schnittstelle erweitert wird.

5.2.3 Low-Level-Lösungen

5.2.3.1 SIP-Servlets

SIP-Servlets sind Java-basierende Programme, die innerhalb eines Java-Webserverns laufen und dort die Anfragen von Clients entgegennehmen. *Servlet* ist ein Kunstwort und besteht aus den Begriffen *Server* und *Applet*. Der Begriff *Applet* wurde von Sun bei der Einführung von Java als ein kleines Programm definiert, was in einem Container (z.B. einem Webbrowser oder einem Webserver) ausgeführt wird [JSR289, JSR315].

Für die serverseitigen Applets stellt der Webserver einen Protokollstack zur Verfügung, über welchen die einzelnen Servlet-Container kommunizieren können. Für SIP-Servlets wurde der Protokollstack um die Schnittstelle zu SIP erweitert. Eine typische Umgebung, in welcher Java Servlets eingesetzt werden ist JavaEE, was als Middleware-Lösung eine große Verbreitung hat.

5.2.3.2 Call Processing Language (CPL)

Die *Call Processing Language* (CPL) ist eine von der IETF spezifizierte XML-basierende Sprache zur Beschreibung von Multimediadiensten und Internettelefonie [RFC3880]. Die Sprache ist unabhängig von Betriebssystem, Netzwerktechnologie oder Signalisierungsprotokollen.

Der Ablauf einer CPL-Anfrage kann als eine Folge von Knoten betrachtet werden, die eine Entscheidung über die nachfolgende Aktionen treffen (Abb. 5.11). CPL ist darauf ausgelegt, dass alle Anfragen und Aktionen grafisch als Knoten mit Ein- und Ausgängen dargestellt werden können. Die möglichen Aktionen werden vom Befehlsatz vorgegeben, können aber auch erweitert werden.

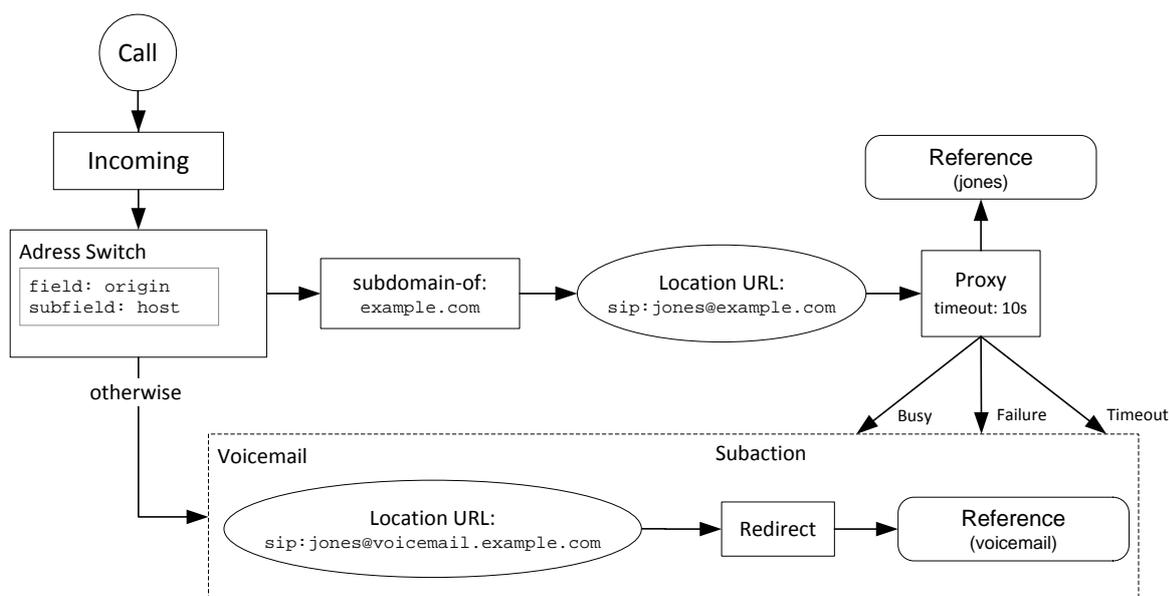


Abbildung 5.11: Ablauf einer Anfrage der Call-Processing-Language zur Entscheidung über Aktionen über eine eingehende Verbindung (nach [RFC3880])

5.2.3.3 SIP-CGI

Das *Common Gateway Interface* (CGI) hat sich zu einem weitverbreitetem Standard für den Datenaustausch zwischen Webserver und Anwendungen zur Erzeugung von dynamischen Webseiten entwickelt. Aufgrund der Ähnlichkeiten zwischen SIP und HTTP, wurde von der IETF eine spezielle Variante des CGI's für SIP spezifiziert [RFC3050]. Da CGI sprachunabhängig ist, kann die CGI Schnittstelle mit allen Programmier- und Skriptsprachen verwendet werden.

5.2.4 Zusammenfassung und Bewertung

Es existiert eine Reihe von Konzepten, die High-Level-APIs mit einer Menge von Funktionen und Abläufen zur Realisierung von Diensten im NGN definieren. Diese APIs sind theoretische Spezifikationen, welche erst von konkreten Implementierungen umgesetzt werden müssen.

Die Low-Level-APIs basieren alle auf bereits bekannten Technologien, die im Internet schon verbreitet sind, wodurch deren Umsetzungen deutlich einfacher ist und teilweise schon im Einsatz ist.

Die größte Verbreitung haben derzeit Produkte auf der Basis der Java-Implementierung der Parlay-Spezifikation, *JAIN SLEE* (z.B. mobicents, OpenCloud Rhino, jNetX), und der *SIP-Servlets* (z.B. mobicents, jNetX, SailFin). Es existieren jedoch auch Produkte,

welche auf anderen High-Level-Anwendungen aufsetzen, und beispielsweise bestehende Anwendungsserver für Webanwendungen um eine eigene IMS-Schnittstelle erweitern.

Diese Implementierungen von Anwendungsservern für NGNs werden nun etwas genauer untersucht und auf einen möglichen Einsatz als Dienstplattform bewertet.

5.3 Verfügbare Softwarelösungen

5.3.1 Frauenhofer Open Source IMS Playground

Der *Open Source IMS Playground* ist eine Beispielimplementierung eines IMS-basierenden Kommunikationsnetzes von der Frauenhofer Arbeitsgruppe FOKUS. Dieses Netz beinhaltet alle Komponenten der Vermittlungs- und Anwendungsschicht des IMS-Modelles (siehe Kapitel 3.1.2). Die Basis bildet der *Open Source IMS Core* (s. Abb. 5.12), die erste Implementierung der Sitzungssteuerungs- und Vermittlungsfunktionen eines IMS-basierenden Netzes. Das Ziel des Projektes ist ein IMS-basierendes Netz

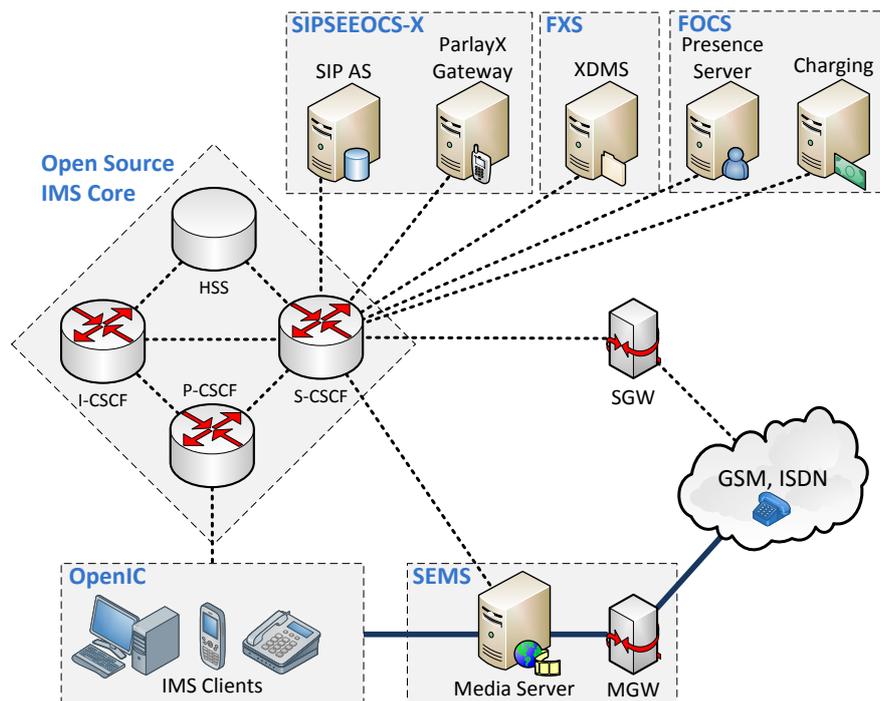


Abbildung 5.12: Komponenten des Open Source IMS Playgrounds, einer kompletten Beispielimplementierung für alle Komponenten des IMS (nach [WeVi08]).

für Forschungszwecke und zum Testen von Service-Konzepten im NGN. Als potentielle Nutzer sind dabei Telekommunikationsanbieter ebenso wie Forschungseinrichtun-

gen vorgesehen. Das Projekt umfasst alle Netzelemente eines IMS-basierenden NGN, wie Anwendungsserver, Medien- und Signalisierungsgateways, bis hin zu IMS-Clients [Frau07]. Die Komponenten sind teilweise als eigene Opensource-Anwendungen verfügbar und teilweise auch als kommerzielle Produkte von Projektpartnern eingebunden.

Die *SIP Servlet Execution Environment* (SIPSEE) ist eine Plattform für einen Java-basierenden Anwendungsserver, der eine SIP-Sitzungssteuerung und verschiedene andere Protokolle wie HTTP oder Diameter mitbringt. Dabei kann auf die Funktionen des *Open Communication Server* (OCS) zurückgegriffen werden, der die Parlay-X-Schnittstelle als Webservice bereitstellt und so eine Beeinflussung der Sitzungssteuerung ermöglicht. Als Anwendungsserver ist auch eine kommerzielle JAIN-basierende Anwendung des Projektpartners *jNetX* vorgesehen [WeVi08]. Diese Plattform integriert die JAIN SLEE und besitzt Schnittstellen zur Kooperation mit anderen Servern, die auf SOA, OSA/Parlay AS oder dem Intelligenten Netz basieren [jNet09].

Auf der Basis des Open Source IMS Playground wurde ein weiteres Projekt gestartet, das als Testumgebung Dienste auf Basis einer *Service-Oriented Architecture* (SOA) dienen soll, der *Open SOA Telco Playground* [Frau09].

5.3.2 IBM WebSphere IMS Connector

Der *WebSphere Application Server* von IBM ist ein kommerzieller, auf JavaEE basierender Server für Webanwendungen. Dieser Server kann durch den *WebSphere IMS Connector* um SIP- und IMS-Funktionalitäten erweitert werden, so dass auf der bestehenden Plattform Dienste für das NGN angeboten werden können [IBM09].

Der Vorteil dieser Lösung ist, dass bereits bestehende Webservices zu Diensten im NGN erweitert oder mit diesen verbunden werden können. Außerdem kann so bereits vorhandenes Know-How zur Websphere-Plattform weiterverwendet werden, um Dienste im NGN zu entwickeln. Der IMS-Connector für Websphere greift dabei auf alle Funktionen zurück, die in der Parlay-X-Schnittstelle definiert sind. Somit können die Dienste neben den SIP-Verbindungen auch auf die IMS-Sitzungssteuerung, Abrechnungsfunktionen oder Nutzerprofile im IMS-Kernnetz zugreifen. Die Plattform stellt dabei auch sogenannte IMS-Enabler bereit, die Dienste wie Präsenzserver enthalten.

Die Lizenzierung des Websphere Application Servers und des Websphere IMS Connectors erfolgt über sogenannte *Processor Value Units* (PVU), die abhängig von der Anzahl und der Rechenleistung der verwendeten CPU-Kerne den Preis bestimmen. Für ein System mit einem Single-Core-Prozessor betragen die Lizenzkosten mindestens 83.300 €.

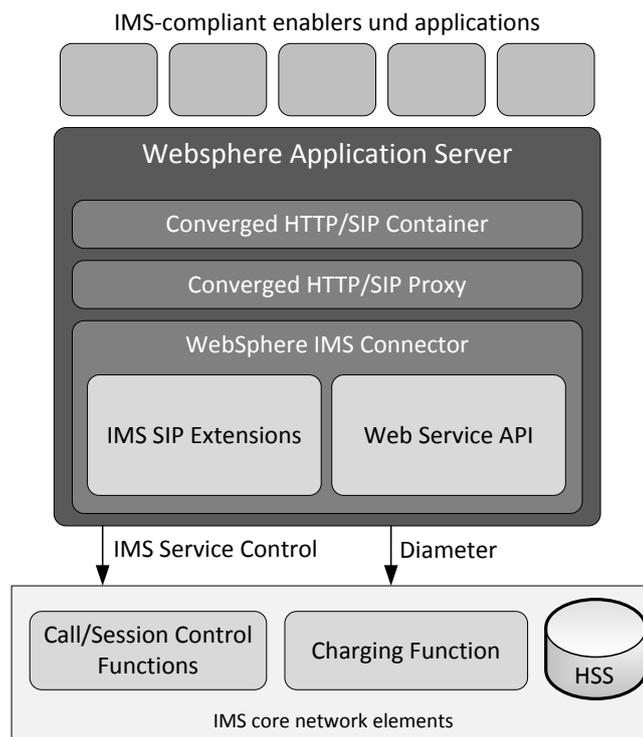


Abbildung 5.13: Der Websphere IMS Connector stellt für den verbreiteten Webserver eine Schnittstelle zum NGN bereit (nach [CSSW07]).

5.3.3 Mobicents

*Mobicents*¹ ist eine Opensource-Plattform zur Erzeugung von Diensten im NGN, welche aus verschiedenen Teilprojekten besteht. Es war die erste Opensource-Implementierung des JAIN SLEE Standards 1.0 [JSR22], unterstützt aber mittlerweile bereits die Funktionen der Version 1.1 [JSR240]. Das Projekt basiert auf dem *JBoss Application Server*, der eine Middleware-Implementierung der Java Enterprise Edition ist.



Abbildung 5.14: Die Mobicents Plattform besteht neben dem SIP-Servlet-Container und der JAIN SLEE aus einem Media- und Präsenzserver.

Mobicents beinhaltet eine *JAIN Service Logic Execution Environment* (JSLEE), einen Server für SIP-Servlets nach [JSR289], einen Media-Server und einen Präsenzserver.

¹<http://www.mobicents.org/>

Durch die Portabilität von JBoss sind Mobicents und die darauf aufbauenden Anwendungen plattformunabhängig. Im Oktober 2009 wurde zusätzlich eine auf Mobicents basierende IP-Telefonanlage (IPBX) veröffentlicht [Rale09].

Die Kommunikation der Anwendungen erfolgt über sogenannte *Ressource Adapter* (RA) [Sila08], welche beispielsweise für SIP, Diameter, XMPP oder HTTP mitgeliefert werden und eine flexible Erweiterung des Anwendungsservers ermöglichen. Es existieren auch spezielle Ressource Adapter zu einer Text-to-Speech-Engine und zur Java-Schnittstelle der IP-Telefonanlage Asterisk (siehe Kapitel 6.3.1).

5.3.4 OpenCloud Rhino Application Server

Der *OpenCloud Rhino Application Server* [Open08] ist ebenfalls ein Anwendungsserver auf Basis der JAIN SLEE 1.1 Spezifikation. Der Anwendungsserver ist der Teil einer Produktreihe für Netzbetreiber, die unter anderem einen grafischen Service-Editor (Rhino Service Interaction Server) und einen Abrechnungsserver (Rhino Charging Sentinel) umfasst.

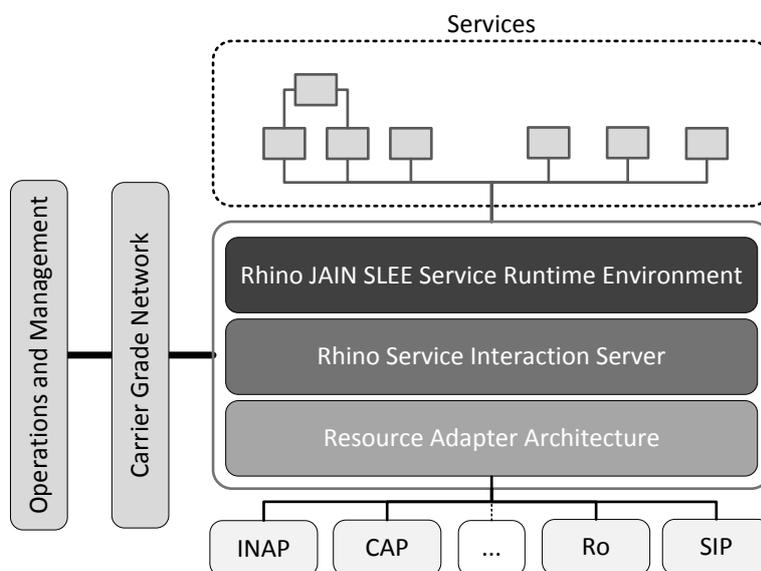


Abbildung 5.15: Der OpenCloud Rhino Application Server ist eine JAIN-SLEE-Plattform und verfügt dementsprechend über eine Ressource Adapter Struktur (nach [Open08]).

Der Anwendungsserver nutzt eine ähnliche Ressource-Adapter-Architektur wie Mobicents, um die einzelnen Protokolle wie SIP, HTTP oder LDAP nutzbar zu machen. Ein Schwerpunkt liegt dabei auf den Protokollen des IN (siehe Kapitel 2.2), wie das SS7 und INAP, um die Dienste in bestehenden Leitungsgebundenen Netzen über

ein IMS-Kernnetz anzubieten.

5.3.5 Ericsson IMS/ SailFin

Der Netzausrüster Ericsson hat in Japan bereits im Jahr 2006 das erste IMS-Netz in Betrieb genommen [Eric06]. Ericsson bietet dafür ein IMS-Kernnetz mit verschiedenen Basisdiensten und einige erweiterte Anwendungen an [Eric08].

Zur Erzeugung neuer Dienste stellt Ericsson das *Service Development Studio* bereit, welches auf einer JavaEE-Umgebung basiert. Als Schnittstellen zum NGN werden hier allerdings die High-Level-Schnittstellen aus dem IMS für Geräte [JSR281] und die Steuerung des Netzes [JSR325] verwendet. Für die Nutzung von SIP-Servlets greift Ericsson auf den Anwendungsserver *SailFin* zurück.

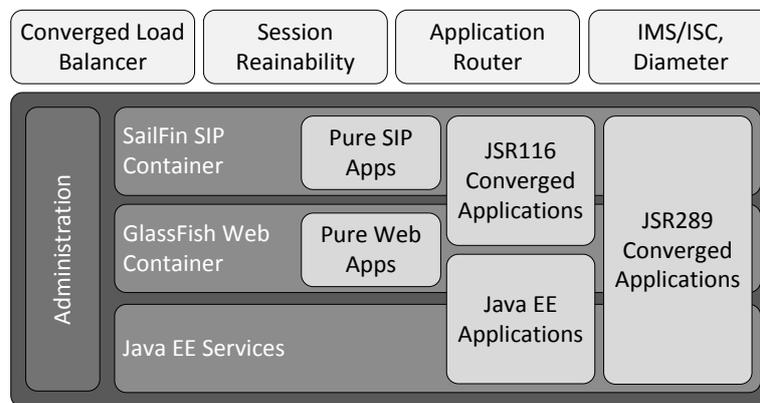


Abbildung 5.16: SailFin ist ein Java-basierender NGN-Anwendungssserver für SIP-Servlets und basiert auf dem WEbanwendungssserver GlassFish (nach [Duvu08]).

Der Opensource-Anwendungssserver *SailFin* stellt im NGN eine Umgebung für SIP-Servlets nach [JSR289] zur Verfügung. SailFin läuft in einer JavaEE-Umgebung und nutzt als Basis den *GlassFish Enterprise Server* von Sun.

Als fertige Anwendungen bietet Ericsson etwa IMS Messaging, Präsenzdienste, Push-to-Talk oder eine Rich Communication Suite an [Mari08].

5.4 Client-Seitige Applikationen

Seit einigen Jahren haben sich, neben reinen Webdiensten und den Basisdienstleistungen der Mobilfunknetze (SMS, MMS etc.), besonders Applikationen, die direkt auf den mobilen Endgeräten laufen, stark verbreitet. Mit der massiven Verbreitung von

Geräten, die ein großes Display und eine hohe Rechenleistung haben, sind viele neue Möglichkeiten für die Verwendung der Endgeräte geschaffen worden.

Die Anwendungen sind nicht mehr nur an ein Endgerät gebunden, da sich mittlerweile vier Standardbetriebssysteme für Mobiltelefone durchgesetzt haben (Windows Mobile, MacOS, Google Android, Symbian OS) die jeweils auf einer großen Anzahl von Endgeräten laufen. Für diese Betriebssysteme stehen schon länger Entwicklungsumgebungen zur Verfügung, wodurch bereits eine Vielzahl von Anwendungen verfügbar sind.

Aber erst mit der Schaffung einer zentralen Anlaufstelle für die Beschaffung dieser Anwendungen wurde ein Boom ausgelöst [Döll09]. Über dem *AppStore* von Apple Inc. für das iPhone wurden bereits in den ersten Monaten mehrere 1000 verschiedene Anwendungen bereitgestellt. Für die anderen Betriebssysteme wurden später ähnliche Plattformen angekündigt und teilweise schon realisiert.

Eine Anwendung, die direkt auf dem Endgerät läuft, bietet mehrere Vorteile gegenüber den Anwendungen, die von Anwendungsservern im Netz bereitgestellt werden. So wird zum einen ein grafisches Nutzerinterface möglich, das sonst so vom Anwendungsserver nicht angeboten werden kann. Zum anderen ist die Anwendung auf dem Endgerät gespeichert, müssen weniger Daten übertragen werden, da bloß dargestellte Inhalte über das Netz transportiert werden müssen.

Der Nachteil ist, dass durch die große Anzahl inkompatibler Betriebssysteme ein großer Aufwand zur Bereitstellung eines Dienstes betrieben werden muss. Nutzer, die kein passendes Endgerät mit einem dieser Betriebssysteme verwenden, sind bei der Nutzung eines solchen Dienstes immer ausgeschlossen.

5.5 Bewertung und Auswahl

Die beschriebene Auswahl ist ein beispielhafter Überblick über die Arten von Anwendungsservern, die im NGN eingesetzt werden können (Tabelle 5.1). Besonders interessant sind dabei alle Lösungen, die als Opensource verfügbar sind und so ohne weitere Einschränkungen als Basis für eine Testplattform verwendet werden können. Der Open Source IMS Playground von der Fraunhofer Arbeitsgruppe FOKUS ist mit seinem Ziel, eine komplette Testumgebung für Forschungs- und Testzwecke zur Verfügung zu stellen, gut für eine Plattform zum Test von weiteren Diensten geeignet. Da das vorhandene NGN im FG Kommunikationsnetze weiter genutzt werden soll, kommt als Anwendungsserver jedoch nur eine Lösung in Frage, die nur die Anwendungsschicht implementiert.

Als die zwei vorherrschenden Standards zur Bereitstellung von Anwendungen haben sich die SIP-Servlets [JSR289, JSR315] und JAIN SLEE [JSR22, JSR240] durchgesetzt. Die einzige Lösung, welche die Nutzung beider Technologien ermöglicht, ist die Plattform *mobicents*. Gleichzeitig ergänzt sie die bestehende IP-Telefonanlage Asterisk um einen Medien- und Präsenzserver und besitzt eine Schnittstelle zum Managementinterface von Asterisk.

Prinzipiell besteht die Möglichkeit, dass verschiedene Dienste die SIP-Schnittstelle und die API der IP-Telefonanlage Asterisk (siehe Kapitel 6.3.1) nutzen. Damit vermischt jedoch die Trennung und die Auslagerung der Dienste aus den Netzknoten. Auf diese Möglichkeit wird im Kapitel 6.3.3 näher eingegangen.

Die folgenden Fußnoten gehören zu Tabelle 5.1.

¹einige Teile sind kommerziell

²eine auf Mobicents aufbauende IP-Telefonanlage wurde im Oktober 2009 veröffentlicht [Rale09]

³es existiert eine Portierung nach Windows (AsteriskWin32)

⁴Wenn nur SIP-Servlets verwendet werden, ist eine Verwendung von Tomcat/Apache möglich.

⁵Ist bisher nur teilweise integriert, Status: *testing*

⁶nachrüstbar über Plugin/ externe Anwendung

Produkt	mobicents	WebSphere IMS Connector	Open IMS Core	Rhino AS	SailFin	Asterisk (AGI)
Hersteller	OS	IBM	Frauenhofer	OpenCloud	Sun	OS
Lizenz	GPL	kommerziell	GPL ¹	kommerziell	GPL	GPL
Kosten	-	83.000€	-	k.A.	-	-
Einsatzgebiet	Provider	Provider	Forschung, Betatests	Provider	Provider	PBX
Konzept (NGN Schicht)						
Anwendungsschicht	x	x	x	x	x	(x)
Vermittlungsschicht	(x) ²	-	x	-	-	x
Transportschicht	-	x	-	-	-	-
Plattformabhängigkeiten						
Betriebssysteme	alle	alle	Unix	k.A.	alle	Unix ³
Datenbank	MySQL	DB2/Oracle/ SQL Server	MySQL	k.A.	JavaDB	AsteriskDB (Berkley)
Webserver	JBoss ⁴	WebSphere	JavaEE	k.A.	JavaEE	Apache
Skalierbarkeit	x	x	x	k.A.	x	-
Technologien/ Sprachen						
JAIN SLEE	x	-	-	x	-	-
SIP Servlets	x	-	x	-	x	-
CSE	-	-	x	x	-	-
andere	JavaEE	WebSphere JavaEE	ParlayX	JavaEE	JavaEE	AGI
Protokolle/ Ressource Adapter (Auswahl)						
Asterisk	x	-	-	-	-	x
DIAMETER	x	x	x	x	x	(x) ⁵
LDAP	x	x	x	x	x	-
SIP	x	x	x	x	x	x
XMPP	x	x	x	-	-	-
Integrierte Dienste (Auswahl)						
Media Server	x	x	x	-	-	-
Presence Server	x	x	x	-	-	(x) ⁵
Translation Server	-	x	-	-	-	-
Sprachsynthese	x	x	x	-	(x) ⁶	(x) ⁶
Spracherkennung	-	x	-	-	-	-

Tabelle 5.1: Vergleich der verfügbaren Softwareplattformen für Anwendungsserver
(Fußnoten siehe vorhergehende Seite)

6 Konzept für Netzaufbau

6.1 Ausgangssituation

Im Fachgebiet Kommunikationsnetze (FG KN) der TU Ilmenau existiert bereits ein prototypisches Netz für Voice-over-IP-Kommunikation, welches gewisse FMC-Leistungsmerkmale integriert. Das Netz wurde in den Medienprojekten [GüRu08] und [StMa08] realisiert.

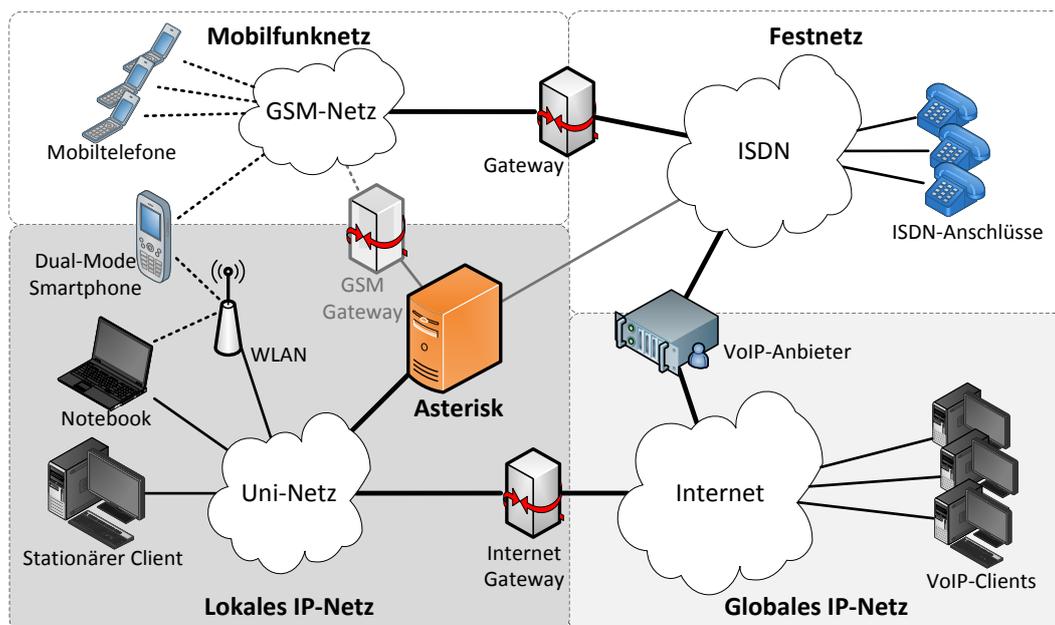


Abbildung 6.1: Aufbau des bisherigen FMC-Netzes, wobei grauen Verbindungen optional sind

In [GüRu08] wurde ein VoIP-Netz auf Basis der IP-Telefonanlage (IP-PBX¹) Asterisk aufgebaut, welches eine eigene Anbindung an das ISDN-Netz besitzt und einige einfache Dienste sowie eine Managementschnittstelle mit Webinterface (FreePBX) besitzt.

¹PBX = Public Branch Exchange (Telefonanlage zwischen lokalem und öffentlichem Netz)

Dagegen wurde in [StMa08] auf die Anbindung des ISDN-Netzes über einen eigenen ISDN-Anschluss verzichtet, und die Anbindung an das Telefonnetz nur über einen VoIP-Provider mit eigenem Gateway realisiert. Um ein Least-Cost-Routing für Gespräche in ein Mobilfunknetz zu ermöglichen, wurde hier zusätzlich ein GSM-Gateway genutzt.

In beiden Konzepten bildet die IP-Telefonanlage Asterisk (siehe 6.3.1) den Mittelpunkt des lokalen Netzes, da Asterisk eine vollständige Signalisierungs- und Sitzungssteuerung ermöglicht. Andere NGN-Komponenten wurden dabei, bis auf das GSM-Gateway, nicht betrachtet, da nur die Realisierung von herkömmlichen Leistungsmerkmalen wie Anrufweiterleitungen oder Konferenzen und einige grundlegende NGN-Leistungsmerkmale wie Sitzungsmobilität gefordert waren.

Die IP-Telefonanlage ist mit dem internen Netzwerk der TU Ilmenau verbunden und dadurch zusätzlich über das drahtlose Netzwerk auf dem Campus und über das Internet via DFN erreichbar. Diese Netze bieten keine QoS-Garantien für Echtzeitdaten, wie es im Konzept für Netze auf Basis von IMS vorgesehen ist. Da die Bandbreite der Netze aber vergleichsweise hoch ist (LAN: 1GBit/s, WLAN: 54MBit/s, Internet-Uplink: 1GBit/s), wird sich dieser Umstand jedoch kaum negativ auf die Qualität der genutzten Dienste auswirken.

6.2 Netzkonzept

6.2.1 Anforderungen

Das prototypische NGN soll die typischen Netzelemente und Funktionen der verschiedenen NGN-Konzepte enthalten. Da die Netzwerkstruktur durch das interne Netz der TU Ilmenau bereits vorgegeben ist und Änderungen dort nur mit großem Aufwand durchführbar sind, können Dienstgütegarantien nur aufgrund überdimensionierter Bandbreite realisiert werden. Techniken zur Einhaltung der QoS-Kriterien (siehe 3.5.1) müssten für eine optimale Funktion in den einzelnen Netzwerkkomponenten (Router, Netzwerkschicht, Access-Points) integriert werden. Über das NGN sollen möglichst viele andere Kommunikationsnetze erreichbar sein, wozu jedoch wie in [StMa08] gezeigt, nicht zwangsläufig Hardware-Gateways nötig sind. Die Anbindung von ISDN und den Mobilfunknetzen kann über einen VoIP-Anbieter mit SIP-Schnittstelle (beispielsweise *sipgate*) realisiert werden. Über die Verbindung zum Uninetz sind auch zwei lokale Zugangnetzwerke verfügbar (Ethernet und WLAN). Von den Zugangnetzwerken anderer Internetanbieter kann über das Internet-Gateway der TU Ilmenau auf das lokale NGN

zugegriffen werden.

Da die Bereitstellung von Mehrwertdiensten in dieser Arbeit ein zentraler Bestandteil ist, muss dafür eine geeignete Plattform eingesetzt werden. In [GüRu08, StMa08] wurden einige Mehrwertdienste bereits auf der Basis von Asterisk implementiert. Da nach den Netzkonzepten für das NGN zur Erzeugung von Mehrwertdiensten ein Anwendungsserver als separate Plattform vorgesehen ist, soll ein solches Element ebenfalls implementiert werden.

6.2.2 Verwendete Netzelemente

6.2.2.1 User Agents

Als UAs kommen verschiedene Softphones und Hardware-Clients zum Einsatz. Die am meisten verbreitete Software für Windows ist die Freeware *X-Lite* (Counterpath). Diese Software wird teilweise von SIP-Anbietern wie sipgate schon vorkonfiguriert ausgeliefert.



Abbildung 6.2: Softphones als User Agents für die Nutzung im NGN (X-Lite, Ekiga, SIP Communicator, v.l.)

Unter Linux ist bei etlichen Distributionen, wie Debian oder Ubuntu, das Softphone *Ekiga* die Standardanwendung für VoIP. Das Programm ist ebenfalls für Windows verfügbar, wobei dort noch Stabilitätsprobleme auftreten. Es existieren auch plattformunabhängige Programme, wie der Java-basierende *SIP Communicator*, die zusätzlich zu den VoIP-Protokollen SIP und H.232 andere IM- und Chat-Protokolle wie ICQ, MSN oder IRC verwenden können. Alle vorgestellten Softphones unterstützen

Video- und Audiodatenströme, und unterscheiden sich nur durch Details des Funktionsumfangs sowie durch die Handhabung. X-Lite ist der einzige dieser UAs, der zwei SIP-Verbindungen gleichzeitig nutzen kann und darüber ohne Konferenzserver eine Dreierkonferenz ermöglicht. Als hardwarebasierende UAs wurden zwei DECT-IP-



Abbildung 6.3: Deutsche Telekom Sinus 501v (links, *Quelle*²) und Planet Communication VIP-320 als hardwarebasierende User Agents

Telefone eingesetzt. Das erste ist ein *VIP-320 VoIP-Router* von Planet Communications, der durch eine niedrige Sprachqualität und die schlechte SIP-Unterstützung nur bedingt nutzbar war. Als zweites Testgerät wurde ein *Sinus 501v* (Abb. 6.3) von der Deutschen Telekom eingesetzt. Beide IP-Telefone unterstützen nur Audiodatenströme. SIP-Nachrichten kann keines der Geräte empfangen. Das Sinus 501v konnte als SIP-Statusnachrichten (Kap. 3.2.2.2) anzeigen. Der VIP-320 VoIP Router beinhaltet jedoch ein überraschendes Feature, denn in der Konfigurations-Webseite ist ein Voice-on-Web-Client (siehe Kapitel 4.3.1) integriert. Über diesen Client kann der Nutzer mit einem Browser-Plugin über die eingerichteten VoIP-Konten und den PSTN-Anschluss des Gerätes telefonieren.

Es existieren auch IP-Telefone, die eine deutlich umfangreichere Unterstützung für die Kommunikation im NGN mitbringen. So bieten beispielsweise Cisco und Snom verschiedene Geräte an, die Videotelefonie und ein Online-Telefonbuch unterstützen. Der Übergang zu Videokonferenzsystemen (siehe Kapitel 4.3.3) ist dabei fließend.

²Deutsche Telekom: Pressebilder <http://www.telekom.com/dtag/cms/content/dt/de/89884>

6.2.2.2 Sitzungs- und Verbindungssteuerung

Für die Sitzungs- und Verbindungssteuerung (CSCF) sind im IMS-Modell getrennte Funktionalitäten für Proxy-, Interrogation- und Server-CSCF vorgesehen. In einem lokal begrenzten Netz werden die Server- und Proxy-Funktionalitäten häufig von einem System, meistens einer IP-PBX (z.B. Asterisk, Freeswitch), bereitgestellt. Die Interrogation-Funktionalität ist nur von Bedeutung, wenn man sich im Netz eines anderen NGN-Providers aufhält und über dessen Zugangssysteme eine Verbindungen unter seiner eigentlichen Netzkennung erhalten möchte. Dieser Anwendungsfall ist in diesem Netzmodell jedoch nicht vorgesehen.

6.2.2.3 Gateways

Die Gateways im NGN können in Elemente der Sitzungssteuerungsebene und der Transportebene aufgeteilt werden. Zur Anbindung eines externen Netzes mit eigenen Signalisierungs- und Sitzungsfunktionen, wie ISDN, werden Gateways in beiden Ebenen benötigt.

Die Fest- und Mobilfunknetze werden das Gateway eines SIP-Providers angebunden, welcher das Signalisierungs- und Mediengateway gegen ein Entgelt bereitstellt. Zur Anbindung anderer Zugangsnetze wird das Internetgateway der TU Ilmenau genutzt, so dass auch Nutzer, die sich über einen DSL-Anschluss oder per UMTS am Registration Server anmelden, die Funktionen des Netzes nutzen können.

6.2.2.4 Registrar Server und HSS

Die Funktionalitäten für den Registrar Server und den Home Subscriber Server stellt in einem lokalen ebenfalls die IP-PBX bereit. Die Auslagerung in externe Systeme ist besonders bei der Erzeugung von Redundanz und bei starkem Verkehrsaufkommen wichtig. Da dieses Netz jedoch nur zur prototypischen Demonstration dient, können diese Funktionalitäten in der IP-PBX erzeugt werden.

Die Nutzerdaten können dabei in den Konfigurationsdateien der IP-PBX gespeichert sein, oder bei einer größeren Nutzeranzahl auch in eine separate Datenbank ausgelagert werden.

6.2.2.5 Anwendungsserver

Zur Erzeugung von Diensten ist im NGN-Konzept ein Anwendungsserver vorgesehen. Die Basisdienstleistungen, wie einfache Kommunikationssitzungen oder Weiterleitungen, werden im NGN bereits durch die Elemente der Steuerungsebene erbracht. Erst

für Mehrwertdienste wird eine Plattform mit einer erweiterbaren Anwendungslogik benötigt, welche die Dienste der Anwendungsschicht bereitstellt. Zu diesem Anwendungsserver können auch weitere Komponenten wie Medien- oder Präsenzserver gehören.

Da sich im NGN die Netzelemente aber auch immer zusammenfassen lassen, können auch innerhalb einer IP-PBX Dienste realisiert werden. Eine genauere Betrachtung dieser Variante ist in Kapitel 6.3.3 zu finden.

6.3 Realisierung

6.3.1 Asterisk PBX als zentrales Element

Asterisk ist eine Opensource-Softwarelösung für Telefonanlagen. Es ist eine modulare Lösung, mit der man Analog-, ISDN- und IP-Telefonie bereitstellen kann. Die einzelnen Funktionalitäten sind miteinander kombinierbar, so dass man die Vorteile jedes Netzes nutzen und miteinander verbinden kann. Da Asterisk auf Standard-PC-Hardware läuft, kann man es ohne weitere Hardware, die das System mit einem ISDN-Netz oder dem analogen Telefonnetz verbindet, erst einmal nur als reine IP-Telefonanlage (IP-PBX) nutzen. Durch die große Verbreitung der Software und der ständig steigenden Nutzung von Telefonie in Paketnetzen wurde Asterisk vor allem im Bereich VoIP und Videotelefonie weiterentwickelt. Es bildet außerdem die Grundlage vieler kommerzieller Telefonanlagen und es existieren Portierungen auf alle gängigen Betriebssysteme und viele proprietäre Plattformen (z.B. fonera-Router, Fritz!Box-Router).

6.3.1.1 Aufbau

Den Kern von Asterisk bildet der sogenannte Asterisk Daemon, der für Bereitstellung aller Kommunikationsfunktionen zuständig ist. Zur Kommunikation mit anderen Servern oder lokalen Programmen stehen verschiedene Schnittstellen bereit. Das *Asterisk Gateway Interface* (AGI) bietet beispielsweise eine Programmierschnittstelle (API) zu allen Programmier- und Skriptsprachen [Wint09a, S.281 ff.] und ermöglicht die Erweiterung von Asterisk ohne Eingriff in den Quellcode. Über das *Asterisk Manager Interface* (AMI) können andere Server oder lokale Programme direkten Zugriff auf die Steuerung der Kommunikation in Asterisk nehmen [Jo⁺ot09],[Wint09a, S. 266 ff.].

6.3.1.2 Kommunikation

Diese Kommunikation kann über alle üblichen Protokolle (SIP, H.232, IAX etc.) stattfinden und auf eine Reihe Codecs (G.711, G.722, GSM etc.) zurückgreifen. Da einige

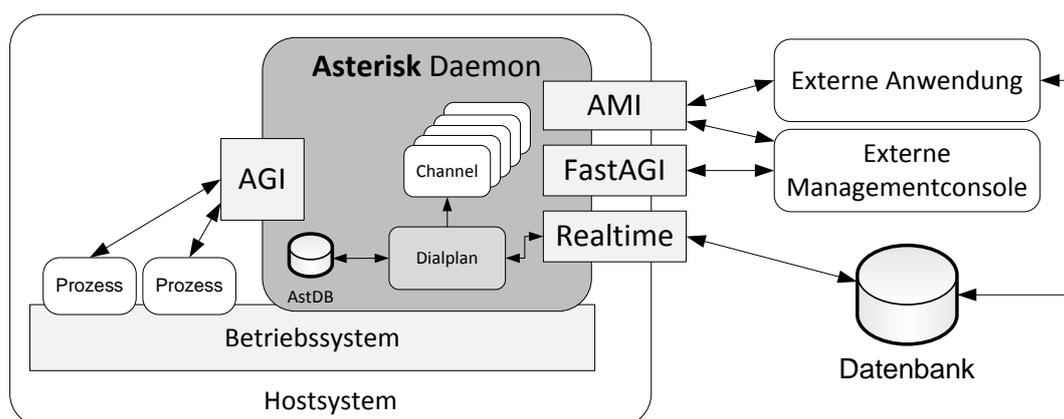


Abbildung 6.4: Interner Aufbau von Asterisk, der Zugriff auf die Kernfunktionen von externen Programmen erfolgt über das Asterisk Gateway Interface (AGI) und auf Verwaltungsfunktionen über das Asterisk Management Interface (AMI).

Codecs proprietäre Lizenzen nutzen, wie beispielsweise G.729, sind diese nicht mit der Lizenz von Asterisk kompatibel und müssen separat bezogen werden. Eine Kommunikationsverbindung zwischen verschiedenen Netzteilnehmern wird innerhalb des Daemons als *Channel* verwaltet.

6.3.1.3 Funktionalitäten

Wie in Kapitel 6.2.2 angedeutet, können einzelne Netzelemente mehrere Funktionalitäten bereitstellen, die im NGN-Modell eigentlich getrennt sind. Die von Asterisk bereitgestellten Funktionalitäten sind:

- Proxy Server
- Registrar Server und HSS (auch über externe Datenbanken)
- Sitzungs- und Verbindungssteuerung (CSCF)
- Gateway (zu ISDN oder POTS mit zusätzlicher Hardware)
- Konferenzserver (nur für Audioverbindungen)
- B2BUA (für Medienkonvertierung)

Damit eignet sich eine Anwendung wie Asterisk sehr gut zum Aufbau eines prototypischen NGN, da nicht alle Netzelemente einzeln eingerichtet werden müssen und alle Funktionalitäten zentral über Asterisk bereitgestellt und gewartet werden können.

6.3.1.4 Dienste

Neben den Basisdienstleistungen für die Kommunikation sind in Asterisk bereits einige Mehrwertdienste integriert. So ist beispielsweise ein Mailbox- und Voicemailsysteem, ein Sprachdialogsystem und ein Konferenzsystem integriert. Über Zusatzprogramme können Sprachsynthese (Text-to-Speech) und Spracherkennung genutzt werden. Für andere Mehrwertdienste kann die AGI-Schnittstelle genutzt werden oder ein externer Anwendungsserver z.B. über SIP angeschlossen werden.

6.3.1.5 Beschränkungen

Bei der Verwendung von Asterisk sind jedoch auch einige Beschränkungen zu beachten. So können die Versionen 1.4 und 1.6 keine SIP-Messages weiterleiten, wenn zwischen den beiden UAs noch kein Channel etabliert wurde [Voip09]. Videoübertragungen sind zwar möglich und werden einfach weitergeleitet, jedoch funktionieren einige Leistungsmerkmale, die von Audioverbindungen bekannt sind, nicht für Videoverbindungen, wie beispielsweise die Konferenzschaltung oder die Medienkonvertierung.

6.3.1.6 Alternative PBX-Software

Neben Asterisk existiert noch weitere Software, die eine ähnliche Funktionalität bietet.

FreeSWITCH ist ein Fork³ aus dem Asterisk-Projekt [Mine08]. Es unterscheidet sich von Asterisk besonders darin, dass jeder Channel auf Betriebssystemebene einen eigenen Thread darstellt. Damit sollte besonders das Verhalten bei der Übergabe und beim Wechsel von Channels verbessert werden, das unter Asterisk in bestimmten Konstellationen zu einem Deadlock führen kann. Den Kern bildet nur noch die Core API, welche die Kommunikation der einzelnen Threads miteinander koordiniert. Unter Asterisk werden alle Channels innerhalb des Asterisk-Daemons ausgeführt. *FreeSWITCH* bietet noch nicht den kompletten Funktionsumfang von Asterisk, ist aber bei den Kernfunktionalitäten durch die veränderte Architektur teilweise stabiler und schneller.

SIP Express Router (SER) ist ein Server der nur SIP-Kommunikation unterstützt und von der Fraunhofer FOKUS Arbeitsgruppe entwickelt wurde. Heute wird das Projekt von iptel.org weiterentwickelt. SER ist ebenso ein Opensource-Projekt, wird aber heute auch von VoIP-Providern eingesetzt, da es gleichzeitig eine große Anzahl von Verbindungen unterstützt. Im Gegensatz zu Asterisk enthält SER nur die Funktionalitäten zur Verbindungs- und Sitzungssteuerung und entspricht so einer Vermittlungstelle für

³Abspaltung in einem Softwareprojekt unter Beibehaltung von Teilen des Quellcodes

SIP-Gespräche ohne Funktionalitäten für Konvertierung von Medienströmen. Parallel dazu existieren weiter Opensource-Projekte wie Kamailio und OpenSIPS (früher OpenSER), die eine ähnliche Funktionalität bereitstellen.

6.3.2 SipToSis - Anbindung an Skype

Eines der heute größten VoIP-Netzwerke ist *Skype*. Dieses Netzwerk basiert auf einem proprietären Client, der für gängige Desktop-Betriebssysteme (Windows, Linux und Mac OS) sowie einige Handyplattformen (Windows Mobile, Symbian) verfügbar ist. Zwischen den Nutzern können auf Basis eines Peer-to-Peer-Netzwerkes Audio- und Videodaten, Textnachrichten und Dateien übertragen werden. Bei neueren Versionen ist auch der Zugriff auf eine proprietäre API möglich, über welche externe Programme Verbindungen zu anderen Nutzern aufbauen können oder in die Kommunikation eingreifen können. Skype ist ein geschlossenes Netzwerk, hat aber besonders durch die kreative Umgehung von Firewalls [Schm06] eine große Verbreitung gefunden. So sind 2009 durchschnittlich 15-20 Millionen Nutzer gleichzeitig online. Besonders in Afrika und in Lateinamerika nutzen 30 bzw. 18% der Internetnutzer den Skype-Dienst [Zieg09]. Für die meisten Länder bietet Skype mit *SkypeOut* außerdem ein kostenpflichtiges Gateway zu den nationalen Telefon- und Mobilfunknetzen an.

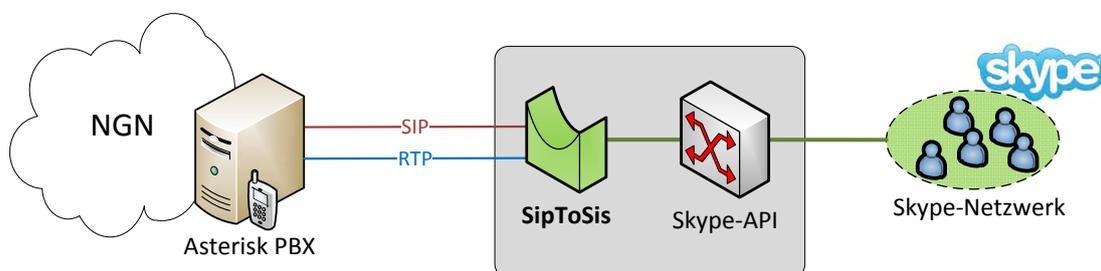


Abbildung 6.5: SipToSis als Gateway zum Skype-Netzwerk

Es existieren bereits einige Produkte, welche die Anbindung von Skype an ein SIP-Netzwerk ermöglichen [digi09]. Im November 2009 wurde von Skype Inc. selbst ein kostenpflichtiges Sykpe-SIP-Gateway angekündigt [Klaß09]. Mit *SipToSis* ist jedoch die erste Opensource-Lösung verfügbar [MHSp09]. SipToSis basiert auf der Java-API *Skype4Java* und funktioniert als B2BUA, da es den UA im Skype-Netzwerk mit einem SIP-UA verknüpft. Es braucht zwingend einen Skype-Client, der die Verbindung zum Skype-Netzwerk herstellt. Da der Client generell nur mit GUI starten kann und unter Linux ab Version 2.0 den Parameter für die Ausgabe auf einem beliebigen (Dummy-) Desktop entfernt wurde, kann Skype nicht mehr ohne Login automatisch gestartet wer-

den.SipToSis kann sich als SIP-UA bei einem Registrar Server (z.B. Asterisk) anmelden, so dass andere Nutzer SipToSis unter seiner ständigen SIP-URI erreichen können.

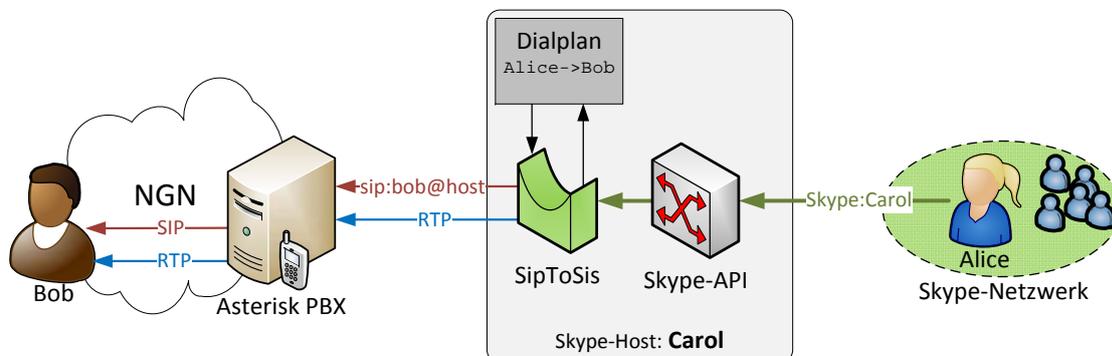


Abbildung 6.6: Eingehendes Gespräch über SipToSis, der Nutzer *Alice* kann *Bob* nicht direkt kontaktieren, sondern muss vom Skype-Host (*Carol*) vermittelt werden.

Intern verwendet SipToSis einen eigenen Dial-Plan, um eingehende Gespräche aus dem Skype-Netzwerk je nach Anrufer zu einer bestimmten SIP-URI weiterzuleiten. Um einen bestimmten SIP-Gesprächspartner direkt zu erreichen, müsste Asterisk das eingehende Gespräch aus dem Skype-Netzwerk annehmen und der Anrufer dann z.B. über DTMF-Töne eine Durchwahl aktivieren.

In umgekehrter Richtung kann der SIP-Nutzer direkt im Dialplan von SipToSis hinterlegte Kontakte anrufen. Diese werden im Dialplan so gespeichert, dass man sie auch über eine numerische ID mit Zifferntasten anwählen kann.

SipToSis kann außerdem auch als PSTN-Gateway genutzt werden, indem die SkypeOut-Funktion genutzt wird, über die man über das Skype-Netzwerk auch in das öffentliche Telefonnetz telefonieren kann. Durch die Konvertierung der Medien und der Signalisierung können Skype-Nutzer anschließend auch auf Mehrwertdienste des lokalen NGN zugreifen. Richtet man einen speziellen Skype-Zugang für den Anwendungsserver ein, können die Nutzer im Skype-Netzwerk direkt mit ihm kommunizieren.

6.3.3 Anwendungsserver vs. Asterisk API

Um einen Mehrwertdienst für die Nutzung mit SIP-UAs zu erzeugen, reicht prinzipiell die Nutzung der API von Asterisk aus. Asterisk selbst stellt darüber jedoch nur einen begrenzten Funktionsumfang bereit, da der Aufgabenbereich von Asterisk primär in der Sitzungssteuerung liegt. D.h. eine „richtige“ SIP-Schnittstelle mit dem vollen Funktionsumfang wurde mit AGI nicht realisiert. Hinzu kommt, dass alle Prozesse, die auf

die Schnittstellen AGI und AMI zugreifen dürfen, die gleichen Benutzerrechte haben. In Asterisk herrscht in dieser Beziehung eine „Anarchie“. Das bedeutet, dass jeder Dienst auch alle Gespräche beenden oder Teile der anderen Dienstleistungen deaktivieren kann.

Somit ist Asterisk für die Demonstration von bestimmten Funktionalitäten geeignet, aber ein Anwendungsserver bietet deutlich mehr Sicherheit und eine größere Auswahl an Funktionen. Von besonderem Interesse sind dabei die Schnittstellen für die Erzeugung von Diensten, die in Kapitel 5.2 vorgestellt werden. Für diese Schnittstellen existieren bereits Implementierungen, die als Anwendungsserver in einem NGN eingesetzt werden können. Bei den in Kapitel 5.3 vorgestellten Softwareplattformen wurde Mobicents als die am besten geeignete Lösung für das prototypische Netz ausgewählt. In Kapitel 7 werden Konzepte zur Realisierung von Mehrwertdiensten für beide Varianten vorgestellt, über die Asterisk API und über einen Anwendungsserver.

6.4 Prototypisches Netz

Als Netzmodell ergibt sich damit ein einfaches 3-Schichtiges Netz, wobei im lokalen Netz jede Netzebene vereinfacht von einer Software zur Verfügung gestellt wird (Abb. 6.7).

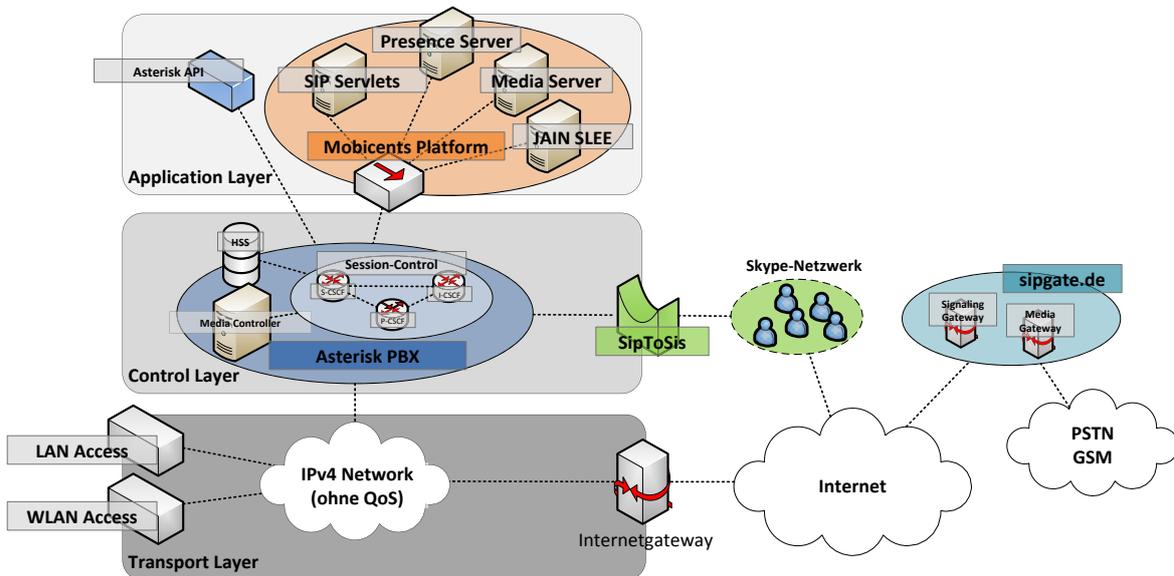


Abbildung 6.7: Das Konzept des prototypischen NGN dargestellt als 3-schichtiges Netzwerk, wobei die Funktionalitäten der Anwendungs- und Steuerungsschicht von monolithischen Anwendungen erzeugt werden.

Die Transportschicht entspricht in dem prototypischen Netz dem lokalen Netzwerk der TU Ilmenau. Dieses Transportnetz kann über ein kabelgebundenes und ein kabelloses Zugangsnetz mit verschiedenen Endgeräten genutzt werden. Die Aufgaben der Sitzungs- und Verbindungssteuerungsschicht werden vollständig von der IP-Telefonanlage Asterisk erzeugt. Für die Erzeugung von Mehrwertdiensten sind in der Anwendungsschicht zwei Schnittstellen vorgesehen. Einfache Dienste können mit der Asterisk API realisiert werden. Als Anwendungsserver kommt die Mobicents-Plattform zum Einsatz, die neben den Plattformen für JAIN und SIP-Servlets einen Medienserver und einen Präsenzserver bereitstellt.

Über das Internetgateway wird die Nutzung des Netzes über alle anderen Zugangsnetze möglich. Andere Gateways realisieren den Anschluss an das PSTN und das GSM-Netz, sowie eine Verbindung zum Skype-Netzwerk.

7 Praktische Umsetzung

In diesem Kapitel wird die praktische Umsetzung der in Kapitel 6 beschriebenen Plattformen zur Erzeugung von Mehrwertdiensten erläutert. Anschließend werden in Kapitel 7.3 Konzepte für die technische Umsetzung einiger prototypischer NGN-Dienste vorgestellt.

7.1 Asterisk

Die IP-Telefonanlage Asterisk (siehe Kapitel 6.3.1) bildet in dieser Arbeit die Grundlage für die Bereitstellung der Vermittlungs- und Sitzungssteuerung für SIP-Verbindungen. Die Steuerung des Verhaltens von Asterisk erfolgt über einen sogenannten Dialplan. Dieser wird hauptsächlich durch zwei Dateien gesteuert. Die Datei `extensions.conf` enthält den Dialplan, welcher die Verarbeitung von eingehenden und ausgehenden Verbindungen beschreibt (*Extensions*). Mit der Datei `sip.conf` werden die Zugänge der Nutzer reguliert. Diese authentifizieren sich gegenüber Asterisk mit ihrem Benutzernamen, der auch ein Teil ihrer dauerhaften URI (siehe Kapitel 3.2.2.1) ist, und ihrem Passwort. Die Nutzer sind jeweils einem *Context* zugeordnet, über welchen der Dialplan gruppiert wird.

7.1.1 Dialplan

Der Dialplan von Asterisk enthält Anweisungen die von Asterisk ausgeführt werden, sobald eine Nummer gewählt wird. Diese Anweisungen werden als *Extensions* bezeichnet und sind die kleinste Programmeinheit im Dialplan [Wint09a, S.50]. Eine solche *Extensions* besteht aus dem Namen der Extension (normalerweise Zahlen oder reguläre Ausdrücke, es sind aber auch Buchstaben zugelassen), einer Priorität und einer Applikation.

```
exten => Name-der-Extension,Priorität,Applikation
```

Programmierung im Dialplan

Innerhalb des Dialplans können nun komplexe Abläufe für den Aufruf einer Extension definiert werden. Die beschriebenen Extensions sind immer nur für einen Context gültig. Über diesen Context ist eine Rechteverwaltung möglich, so dass beispielsweise nur Nutzer aus einem bestimmten Kontext kostenpflichtige Anrufe durchführen können. Innerhalb der Extension können natürlich auch Extensions anderer Contexte aufgerufen werden.

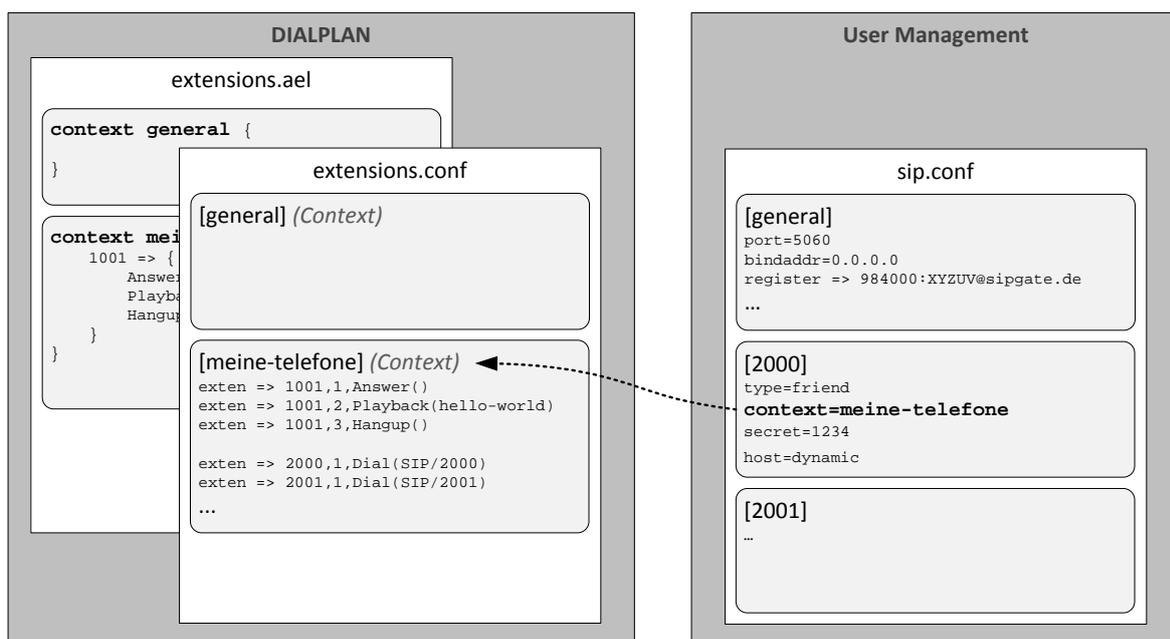


Abbildung 7.1: Der Dialplan in Asterisk bestimmt den Ablauf nach der Eingabe einer Rufnummer. Die Nutzer sind dabei auf ihren jeweiligen Context beschränkt.

Wird eine Extension (z.B. 1001) aufgerufen, werden nacheinander alle Befehle in aufsteigender Reihenfolge der Prioritäten, bei eins beginnend, abgearbeitet (siehe Listing A.1). Dabei wird jeweils die am Ende stehende Applikation aufgerufen. Diese können etwa mit `Answer()` einen Verbindungsversuch akzeptieren oder mit `Playback(hello.gsm)` eine angegebene Audiodatei abspielen. Ein vollständiger Überblick über alle Applikationen befindet sich in [Wint09a, S.421 ff.]¹

Neben den Applikationen existieren im Dialplan auch noch sogenannte Funktionen [Wint09a, S.733 ff.], die beispielsweise String-Operationen ermöglichen. Auf diese Funktionen kann jedoch nur lesend zugegriffen werden. Für Schreibzugriffe ist die Nutzung

¹Die Onlineversion ist unter <http://www.das-asterisk-buch.de/2.1/applications.html> zu finden.

der Funktion `Set ()` nötig.

Listing 7.1: Minimalbeispiel als conf-Datei

```
[meine-telefone]
exten => 1001,1,Answer ()
exten => 1001,2,Playback (hello-world)
exten => 1001,3,Hangup ()
```

Die Nutzung von Variablen ist auch möglich. Diese sind, ähnlich wie in Perl, mit `${Variable}` gekapselt. Auf diese Variablen kann ebenso nur lesend zugegriffen werden, der Schreibzugriff erfolgt wieder über die `Set ()`-Funktion. Es existieren auch einige Systemvariablen, wie `${EXTEN}`, die beispielsweise die Bezeichnung der soeben aufgerufenen Extension beinhaltet. Wenn über reguläre Ausdrücke (z.B. `_200X`) mehrere Extensions zusammengefasst sind, so dass beispielsweise nicht für jede interne Telefonnummer eine eigene Extension aufgelistet sein muss, kann über `Dial (SIP/${Variable})` trotzdem der gewünschte Anschluss erreicht werden.

Programmierung in AEL

Eine etwas übersichtlichere Variante der Programmierung im Dialplan ist die *Asterisk Extension Language* (AEL). Diese Sprache ermöglicht die übersichtliche² Erstellung von größeren Dialplans oder von komplexeren Abläufen mit bedingten Anweisungen und Verschachtelungen.

Listing 7.2: Minimalbeispiel in AEL

```
context meine-telefone {
    1001 => {
        Answer ();
        Playback (hello-world);
        Hangup ();
    }
}
```

Die Regeln des herkömmlichen Dialplans und der AEL-Datei werden vom System automatisch zusammen verwaltet. Es besteht für die Funktionalität also kein Unterschied in welcher Form die Abläufe beschrieben sind.

²AEL bietet keine neuen Funktionen gegenüber der conf-Datei, diese ist aber bereits nach der ersten Verschachtelung kaum noch verständlich.

Über die Dialplan-Applikation AGI () können externe Skripte aufgerufen werden. Diese kommunizieren über eine Schnittstelle mit Asterisk und können zur Implementierung von verschiedene Dienste genutzt werden.

7.1.2 Asterisk Gateway Interface (AGI)

Das *Asterisk Gateway Interface* (AGI) ist eine API für Asterisk, die der CGI-Programmierschnittstelle auf Webservern ähnelt [Wint09a, S.281]. AGI-Skripte können über AGI auf alle Funktionen und Befehle von Asterisk zugreifen, und können prinzipiell in allen Programmier- oder Skriptsprachen entwickelt werden. Voraussetzung dafür ist, dass sie mit Datenströmen unter Linux umgehen können (STDIN, STDOUT, STDERR).

Wird ein AGI-Skript im Dialplan durch eine Extension aufgerufen, bekommt das AGI-Skript über STDIN einen sogenannten AGI-Header übergeben, der alle Informationen über den aufrufenden Channel beinhaltet. Danach arbeitet das Skript diese Informationen ab und kann per STDOUT Befehle innerhalb von Asterisk ausführen. Diese AGI-Befehle unterscheiden sich nur geringfügig von den Befehlen und Applikationen im Dialplan.

Die Skripte selbst liegen im Pfad `/var/lib/asterisk/agi-bin/` und müssen für Asterisk als ausführbar gekennzeichnet sein. Für das Debugging von AGI-Skripten muss Asterisk direkt gestartet werden, da nach einer Verbindung mit einer bestehenden Instanz zwar das AGI-Debugging aktiviert werden kann, die Debug-Ausgaben des Streams STDERR nicht angezeigt werden.

```
# asterisk -vvvvr
(...)
*CLI> agi set debug on
AGI Debugging Enabled
```

Die verwendeten Skripte basieren alle auf der Programmiersprache Perl und können so auf jedem Linux-System verwendet werden. Über die AGI-Schnittstelle können verschiedene Dienste realisiert werden, etwa erweiterte Sprachmenüs, Abfragesysteme für Mails oder Informationssysteme.

7.2 Mobicents Anwendungsserver

7.2.1 Aufbau

Mobicents basiert auf dem JBoss-Anwendungsserver und einer JavaEE-Umgebung (siehe Kapitel 5.3.3). Es existieren vier einzelne Module, die unabhängig voneinander verwendbar sind. Die Logik zur Bereitstellung von Diensten bieten die *JAIN Service Logic Execution Environment* (JAIN SLEE oder JSLEE) und der *SIP-Servlet-Container*. Beide können unabhängig voneinander Dienste bereitstellen und dementsprechend auch einzeln installiert werden. Der Media Server und der Presence Server bieten zusätzliche Funktionalitäten, die für die Erzeugung von Diensten genutzt werden können. Sie können zwar auch einzeln verwendet werden, erweitern aber meistens eine Installation der JAIN SLEE. Wird nur der SIP-Servlet-Container verwendet, kann als Plattform auch

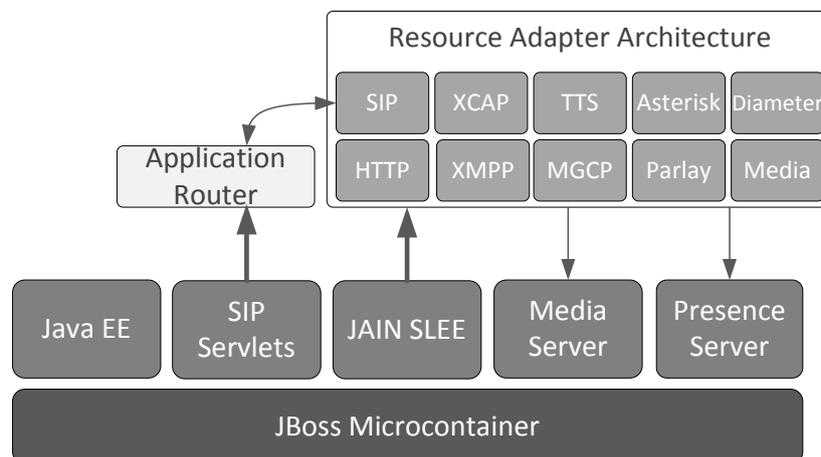


Abbildung 7.2: Die Mobicents Plattform basiert auf dem JBoss-Anwendungsserver, und ist in verschiedene unabhängige Module unterteilt. Die JAIN-SLEE-Plattform verfügt über Resource Adapter zur Kommunikation mit externen Systemen.

Tomcat/Apache verwendet werden. Da jedoch besonders die Kombination der JAIN SLEE mit dem Media-Server für die Erzeugung von Diensten interessant ist, wird hier der JBoss-Anwendungsserver als Basis verwendet.

Für die Kommunikation mit externen System sind in der Spezifikation der JAIN SLEE [JSR22] sogenannte *Ressource-Adapter* (RA) vorgesehen, die jeweils ein Protokoll oder eine Funktion nutzbar machen. Solche RA existieren beispielsweise für die Protokolle HTTP, SIP, Parlay oder Diameter. Der Zugriff auf den Media- und den Präsenzserver der Plattform erfolgen ebenfalls über einen speziellen RA. Da die RAs in der Spezifikation nicht abschließend definiert wurden, hat sie jede JAIN-Implementierung

unterschiedlich umgesetzt [Haig05]. Erst in der JAIN SLEE 1.1 Spezifikation [JSR240] wurde ein einheitlicher Standard geschaffen, so dass nun auch RAs von Fremdanbietern gekauft und eingesetzt werden können. Die Mobicents JAIN SLEE ist konform zur Spezifikation 1.1 und seit dem Erscheinen der Mobicents-Plattform 2.0 im Dezember 2009 erfüllt sie auch die JAIN-SLEE-Spezifikation 2.0.

Die SIP-Servlets selbst können nicht mit RAs umgehen, da die Servlet-Architektur [JSR289, JSR315] nur einen festen Satz Schnittstellen vorsieht. Durch diese festen Vorgaben sind die Anwendung aber gleichzeitig unabhängig von der Mobicents-Plattform. Es besteht aber die Möglichkeit der Kommunikation über SIP-Nachrichten zwischen verschiedenen Diensten, die in der JSLEE und als SIP-Servlet laufen.

7.2.2 Funktionsweise

Die einzelnen Komponenten der JAIN SLEE sind als *Service-Building-Blocks* (SBB) ausgeführt, welche wiederum Grundfunktionen wie SIP-Proxy, Sitzungssteuerung oder Abrechnungsfunktionen bereitstellen. Die einzelnen Dienste kommunizieren untereinander über Events, einem asynchronen Nachrichtentyp. Die SBB selbst bestehen aus *Activity Contexten*, bestimmte Event-Typen empfangen und verschicken können. Ein SBBs kommuniziert nur über diese Activity Contexte und verarbeitet die Nachrichten anschließend intern weiter. Die SIP-Servlets kommunizieren untereinander ebenso

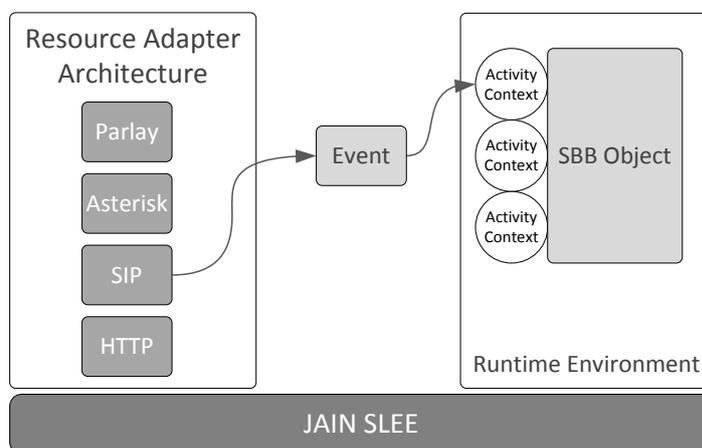


Abbildung 7.3: Dienste der JAIN SLEE sind als Service-Building-Blocks (SBB) ausgeführt, die über einen Activity Context Nachrichten (Events) empfangen können (frei nach [Haig05])

wie mit jedem anderen System im Netz über SIP-Nachrichten. Die SIP-Servlets werden über einen *Default Application Router* gesteuert. Dieser entscheidet anhand der ersten Nachricht einer SIP-Sitzung und den Regeln in einer Konfigurationsdatei, an welche

Anwendung diese Nachricht weitergeleitet wird. Eine solche Regel könnte beispielsweise so aussehen:

```
INVITE: ("org.mobicents.servlet.sip.testsuite.Click2DialApplication",
"DAR:From", "ORIGINATING", "", "NO_ROUTE", "0", "DIRECTION=OUTBOUND"),
("org.mobicents.servlet.sip.testsuite.LocationServiceApplication",
"DAR\ :From", "ORIGINATING", "", "NO_ROUTE", "0", "DIRECTION=INBOUND")
```

In dieser Regel sind für den SIP-Nachrichtentyp INVITE zwei Anwendungen definiert, das Servlet `LocationServiceApplication` wird für SIP-Nachrichten von außerhalb aufgerufen, jedoch nicht wenn die `Click2DialApplication` eine SIP-Sitzung mit einer INVITE-Nachricht von innerhalb aufbauen will. Die gesamte Verwaltung eines SIP-Servlet-Containers besteht hauptsächlich in der Konfiguration dieser Routingregeln.

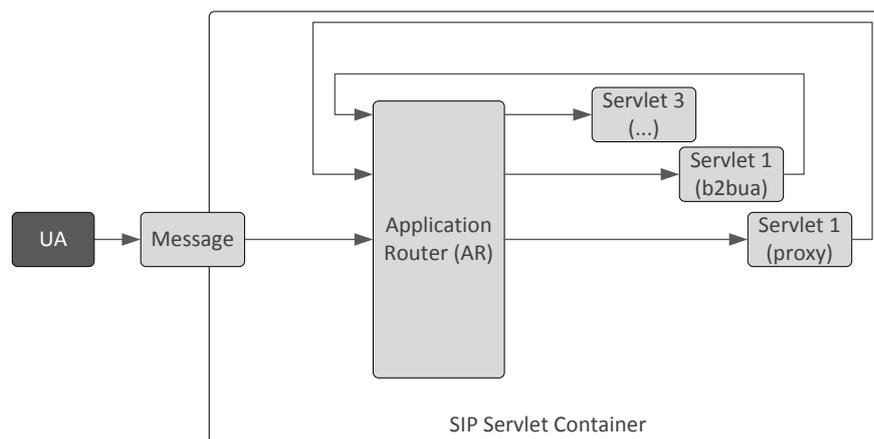


Abbildung 7.4: Der Application Router für SIP-Servlets entscheidet für jede Nachricht anhand ihres Ursprungs und Typs, an welches Servlet sie weitergeleitet wird.

7.2.3 Verwaltung

Die Konfiguration der Mobicents-Plattform erfolgt über ein Webinterface, wobei dieses für jedes Modul (SIP Servlets, JAIN SLEE) jeweils eine eigene Webseite bereitstellt. Diese Seiten sind als Unterfunktionen der Management-Konsole von JBoss angeordnet (Abb. 7.5). Über dieses Webinterface können für die JSLEE-Plattform RAs und Anwendungen hinzugefügt und entfernt werden. Diese Softwarekomponenten liegen als sogenannte *Deployable Unit Files* vor, welche ein automatisches Installationskript beinhalten, das benötigte Komponenten (RAs oder andere Dienste) bei Bedarf eben-



Abbildung 7.5: Management-Konsolen für die Mobicents-Plattform

falls installiert. Für den SIP-Servlet-Container wird hier der *Default Application Router* konfiguriert.

7.3 Dienstkonzepte

7.3.1 Ring Back when Free (RBWF)

Der Dienst *Ring Back when Free* (RBWF) soll den „Rückruf-bei-Besetzt“ mit der Funktionalität des NGN realisieren. Während des Verbindungsaufbaus soll erkannt werden ob das Ziel erreichbar ist. Ist das Ziel zu diesem Zeitpunkt besetzt oder nicht verfügbar, kann auf Wunsch des Nutzers ein automatischer Rückruf gespeichert werden. Danach soll regelmäßig geprüft werden ob der Anschluss wieder frei ist und der Initiator anschließend zurückgerufen werden.

Realisierung

Da die Information über den Status der Verbindung innerhalb der IP-PBX verfügbar ist, kann der Dienst über die Asterisk API realisiert werden. Dazu wird jedem Sitzungsaufbau, der von der Gegenseite mit einer anderen SIP-Statusmitteilung als *200 OK* beantwortet wird, über AGI ein Perl-Skript ausgeführt. Dieses Skript fragt den

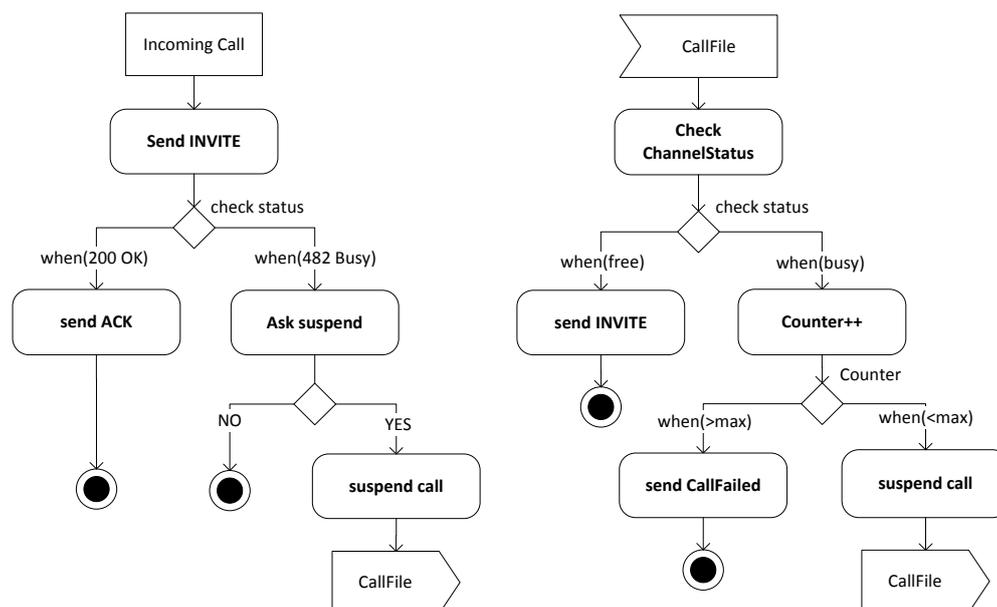


Abbildung 7.6: Aktivitätsdiagramm für Ring-Back-When-Free-Dienst, beim Anlegen eines neuen Rückrufs (links) und bei der Prüfung eines gespeicherten Rückrufs (rechts)

einleitenden Nutzer, ob er eine Benachrichtigung wünscht wenn der gewünschte Gesprächspartner wieder verfügbar ist. Falls das zutrifft, erzeugt das Skript ein CallFile [Wint09a, S.263]. Dieses CallFile wird von Asterisk zu einer vorgegebenen Zeit geöffnet,

und die darin enthaltene Extension aufgerufen. Damit wird das Skript wieder aufgerufen und prüft den Status des gewünschten Anrufers. Für Gespräche zu Teilnehmern die an dieser Asterisk-Instanz angemeldet sind, prüft das Skript die offenen Channels und erzeugt entweder einen Rückruf, falls der Gesprächspartner frei ist, oder erzeugt wieder ein CallFile.

Ausblick

Wenn alle Netzelemente die Abfrage über einen Präsenzserver unterstützen und die Endgeräte ihren Status veröffentlichen, kann bereits vor dem Verbindungsaufbau geprüft werden, ob der Nutzer gerade erreichbar ist. Außerdem kann damit der Status aller Netzteilnehmer ohne einen Verbindungsaufbau geprüft werden, was in der vorgestellten Lösung nur für Nutzer der lokalen Telefonanlage gilt.

7.3.2 Combinatory Services

Über den Combinatory Service (verteilte Sitzung) sollen die Datenströme einer SIP-Sitzung auf mehreren Endgeräten getrennt terminiert werden. Dafür soll ein UA den Videodatenstrom und ein UA den Audiodatenstrom terminieren.

Realisierung

Da eine SIP-Sitzung nur zwischen zwei Endgeräten (UA) etabliert werden kann, muss in diesem Fall der Anwendungsserver die SIP-Sitzung zum entfernten Teilnehmer terminieren, und die lokalen Endgeräte die an der Sitzung beteiligt werden sollen über getrennte SIP-Verbindungen kontaktieren.

Für diese Anwendung eignet sich die Kombination aus der JAIN SLEE und dem Resource Adapter für den Media Server. Die Sitzung wird eingeleitet, indem ein entfernter UA eine Verbindung zu einer URI aufbaut, für die der AS als „Fake-Proxy“³ registriert ist. Er leitet das eingehende Gespräch jedoch nicht weiter, sondern erzeugt als Back-to-Back-UA eine zweite Sitzung zum gewünschten Gesprächspartner. Nimmt dieser die eingehende Verbindung vom AS an, terminiert der „Fake-Proxy“ die SIP-Sitzung selbst, und baut über den Media-Server-RA eine Mediensitzung zum entfernten Gesprächspartner und zu den verschiedenen UAs des lokalen Nutzers auf.

³Ein „echter“ Proxy ist ein Netzelement, welches keine SIP-Anfragen generieren oder Verbindungen terminieren kann.

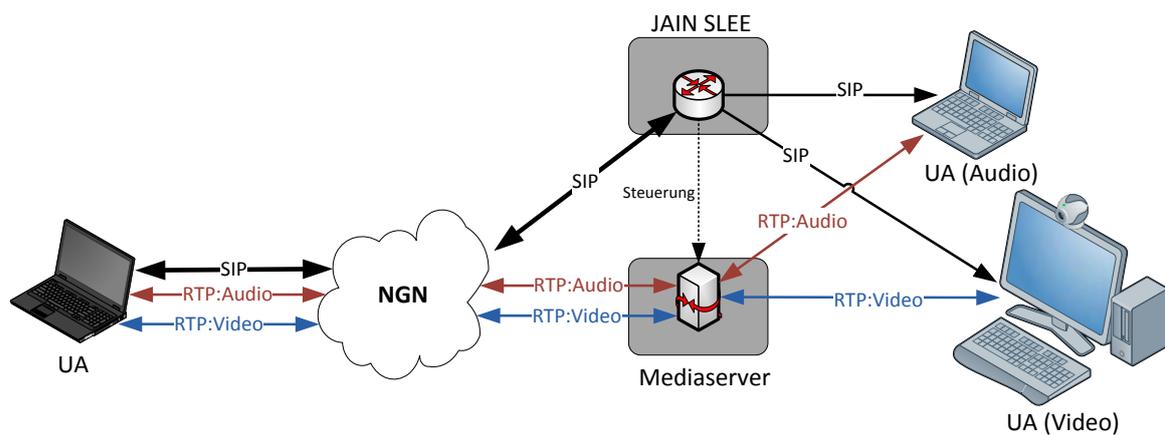


Abbildung 7.7: Combinatory Services erlauben eine Aufspaltung einer SIP-Verbindung mit mehreren Medienströmen auf verschiedene UAs.

Ausblick

Die vorgestellte Lösung enthält ein statisches Routing auf vorher festgelegte Endgeräte mit eigenen URIs. Diese URIs können, wie in Kapitel 3.2.2.1 besprochen, dauerhafte URIs sein, so dass der Nutzer den Dienst von wechselnden UAs nutzen kann. Durch die Adressierung des Anwendungsservers als „Fake-Proxy“ ist die Nutzung dieses Dienstes ohne Anpassung auf der Seite des entfernten Gesprächspartners möglich. Eine sinnvolle Erweiterung wäre die Einbeziehung von Nutzerprofilen wie dem Standort, so dass ein eingehendes Gespräch etwa auf das nächste Videogerät übertragen wird, während der Nutzer den Audiodatenstrom mit seinem Mobiltelefon terminiert. So könnte er ohne ein videofähiges Endgerät im Konferenzraum seiner Arbeitsstelle trotzdem automatisch ein Videogespräch entgegennehmen. Der Übergang zu dem in Kapitel 4.3.1 beschriebenen *Device-unify-Service* ist fließend.

7.3.3 Click-to-Dial

Eine Click-to-Dial-Anwendung soll einen Anruf vom Endgerät des Nutzers initiieren, ohne dass dieser dort ein Ziel angeben muss. Das Gesprächsziel wird in einer externen Anwendung ausgewählt, welche anschließend die Sitzung zwischen dem Endgerät des Nutzers und des entfernten Gesprächspartners einleitet. Die Anwendung kann ein Onlinetelefonbuch, ein zentrales Adressbuch oder ein lokales PIM-Programm wie Microsoft Outlook sein.

Realisierung

Das *Telephony Application Programming Interface* (TAPI) bietet innerhalb des Betriebssystem Windows eine Schnittstelle für Telefonieanwendungen. Dafür existieren Treiber, wie SIP TAPI⁴, die jedem Programm eine SIP-Kommunikation über diese Schnittstelle ermöglichen. Zur Einleitung des Gespräches kann der Nutzer nun in sei-

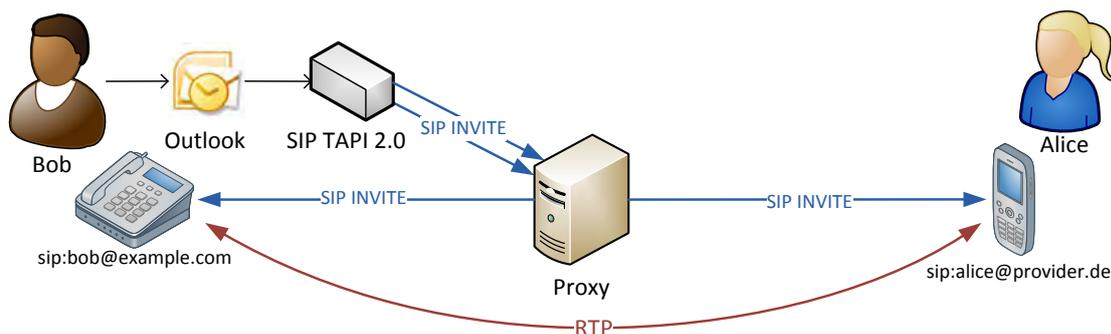


Abbildung 7.8: Click-to-Dial leitet ein Gespräch zwischen zwei Teilnehmern ein, die anschließend direkt miteinander kommunizieren können.

nem lokalen Outlook bei einem beliebigen Kontakt die integrierte Anruf Funktion nutzen (Abb. 7.9 rechts). Daraufhin baut SIP TAPI eine Sitzung zum UA von Bob auf. Die URI des UA muss im Konfigurationsdialog von SIP TAPI hinterlegt sein (Abb. 7.9 links). Nimmt er das Gespräch entgegen, leitet SIP TAPI eine SIP-Sitzung mit seinem ausgewählten Gesprächspartner ein. SIP TAPI funktioniert dabei als B2BUA, da es zwei SIP-Sitzungen einleitet. Die Mediensitzung wird dabei, wie bei SIP üblich, direkt zwischen den zwei Gesprächspartnern etabliert.

Ausblick

Click-to-Dial ist ein Dienst, der an relativ vielen Stellen zum Einsatz kommen kann. Besonders webbasierende Varianten sind häufig zu finden. Eine Beispielanwendung mit einem interessanten Ansatz ist das SIP Servlet *Facebook Click To Call*⁵. Das Servlet funktioniert als Facebook-Plugin, und ermöglicht einen Anruf zu einem anderen Facebookkontakt durch einen Klick auf sein Profil in Facebook. Es etabliert, ähnlich wie SIP TAPI, ein Gespräch zwischen den UAs der Teilnehmer als B2BUA über den SIP-Servlet-Container.

⁴<http://sourceforge.net/projects/siptapi/>

⁵<http://www.mobicens.org/facebook-c2c.html>

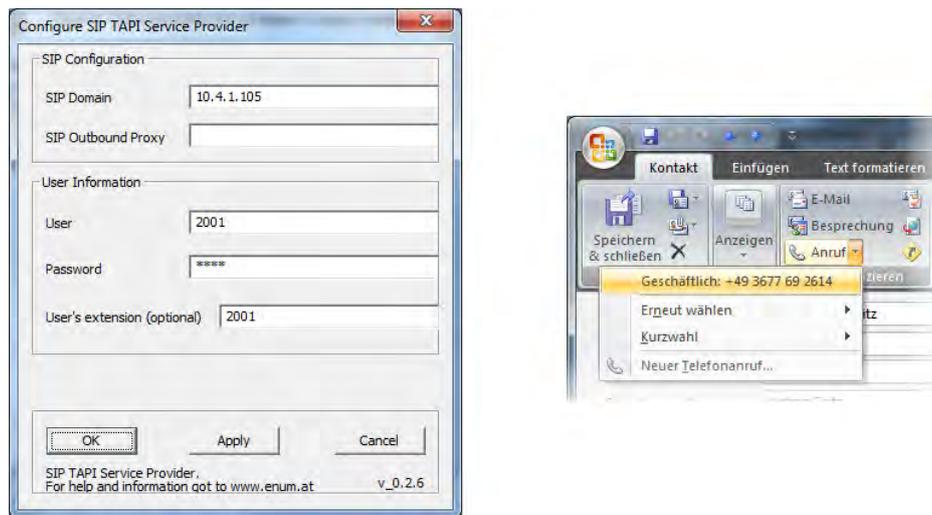


Abbildung 7.9: SIP TAPI Einrichtung (links), Einleitung eines Gesprächs zwischen dem eigenen UA und dem Gesprächspartner über Microsoft Outlook.

7.3.4 Mediencontroller

Der Mediencontroller soll die Fernsteuerung einer Medienwiedergabe über ein SIP-Endgerät ermöglichen. Wie in Kapitel 4.3.7 beschrieben, stehen dafür prinzipiell zwei Ansätze zur Verfügung - die direkte Steuerung des Wiedergabegerätes oder die indirekte Steuerung des Videodatenstromes über einen Mediaserver.

Realisierung

Um den Gedanken der entfernten Steuerung von Geräten umzusetzen, wird hier der Wiedergabeclient ferngesteuert. Ein geeignetes Programm für die entfernte Steuerung ist der Mediaplayer *VLC*⁶, welcher eine Vielzahl von nicht-grafischen Kontrollinterfacies hat. Für eine entfernte Steuerung kommen etwa das HTTP-, das CORBA- oder das Telnet-Interface in Frage. Die Steuerung soll über vorher definierte DTMF-Töne erfolgen. Dafür kann ein AGI-Skript eingesetzt werden, so dass der Nutzer eine spezielle Extension in Asterisk anwählt, die sich anschließend mit dem Mediaplayer per Telnet verbindet. Während der Nutzer mit dieser Extension in Asterisk verbunden ist, kann er die Wiedergabe steuern. Da VLC nicht nur ein Mediaplayer, sondern auch ein Streaming-Server ist, kann das Video nicht nur auf dem lokalen Monitor wiedergegeben, sondern auch per RTP auf einen anderen Client übertragen werden.

⁶<http://www.videolan.org/vlc/>

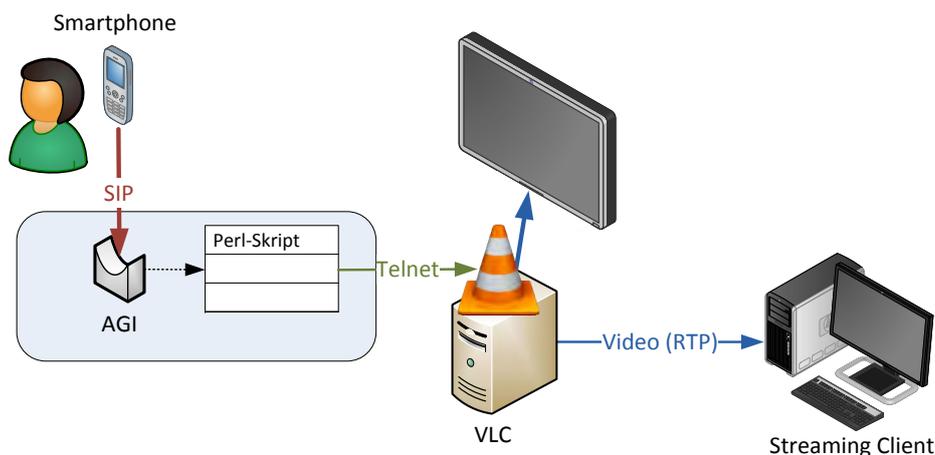


Abbildung 7.10: Mit dem Mediencontroller-Dienst kann über jedes SIP-Endgerät die Wiedergabe auf einem Streaming-Client beeinflusst werden.

Ausblick

Der Dienst kann prinzipiell an zwei Stellen erweitert werden. Zum einen könnte der Nutzer, statt nur per DTMF-Tönen die Wiedergabe zu steuern, über ein Menü mit einem Sprachsynthesystem aus einer Medienbibliothek eine Datei zur Wiedergabe auswählen. Andererseits wäre bei einer Implementierung auf einem Anwendungsserver mit einer leistungsfähigen Spracherkennung eine Fernsteuerung der Medienwiedergabe mit natürlicher Sprache denkbar. Der Nutzer könnte dann sein mobiles Endgerät zur Steuerung im Freisprechmodus auf den Tisch legen, und dann beispielsweise durch einen Zuruf „Pause!“ die Wiedergabe anhalten, wenn er den Raum verlässt.

Im Sinne der in Kapitel 4.3.7 beschriebenen Hausautomation ist, zusätzlich zur Fernsteuerung der Medienwiedergabe, die Verbindung mit der Steuerung der Umgebungsbedingungen (Beleuchtung, Rollläden etc.) vorstellbar. Wobei diese nicht nur auf offensichtliche Zusammenhänge (Licht dimmen bei Filmwiedergabe) beschränkt sein muss, sondern zusätzlich die Kontextinformationen aus den Profilen der Bewohner einbeziehen kann (bei lauter Wiedergabe von Heavy-Metal die Türen zum Lärmschutz schließen) oder die Bewohner auch explizit Benachrichtigen kann (im Wohnzimmer läuft gerade die *Lindenstraße*) falls sie regelmäßig Fernsehprogramme konsumieren.

7.3.5 IPTV Call Notifier

Der IPTV Call Notifier soll eine Benachrichtigung über eingehende SIP-Sitzungen während einer Videowiedergabe ermöglichen, ohne dass sich ein anderes SIP-Endgerät in Reichweite des Nutzers befindet.

Realisierung

In Kapitel 4.3.3 wurde bereits die Problematik des Eingriffs in den Videodatenstrom diskutiert. Ein Mediaserver im NGN kann bei einer Multicastverbindung für die IPTV-Wiedergabe nicht in den Videodatenstrom eingreifen und in das Videosignal eines einzelnen Clients einen Hinweis einblenden. Wenn der Multicastdatenstrom lokal von einem Mediaserver empfangen wird und von dort zum Wiedergabegerät weitergeleitet wird, ist eine Manipulation des Datenstroms möglich. Um eine Benachrichtigung über

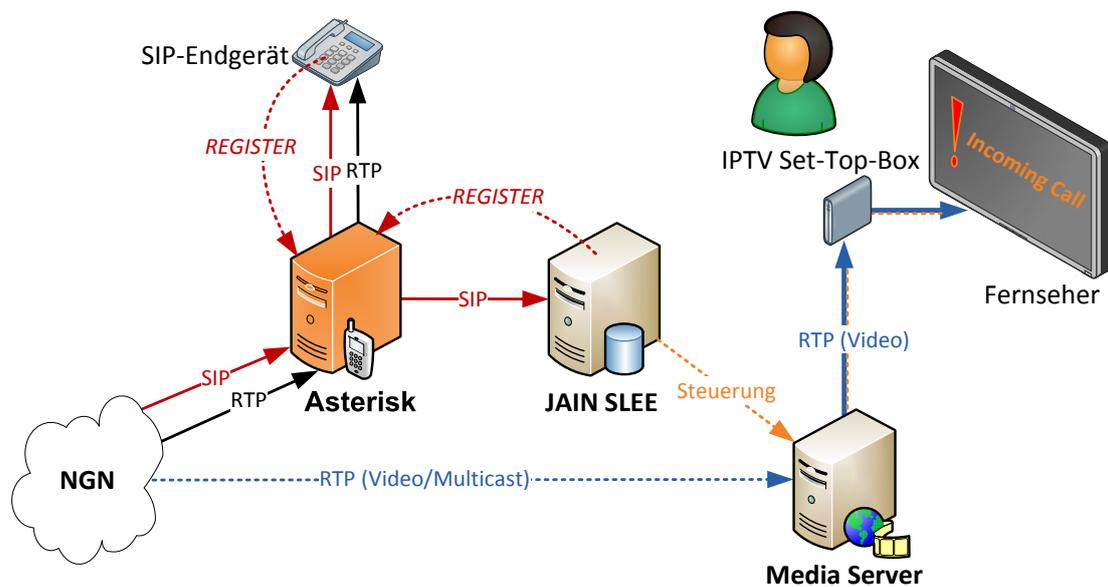


Abbildung 7.11: IPTV-Call-Notifier

ein eingehendes Gespräch zu erhalten, muss sich der Anwendungsserver für die gleiche dauerhafte URI registrieren, wie der UA, der die Gespräche entgegennehmen soll. SIP unterstützt die Registrierung von mehreren temporären URIs für eine dauerhafte URI. Ob die eingehenden Gespräche anschließend aber an alle UAs gleichzeitig weitergeleitet werden hängt jedoch vom Anbieter ab [TrWe07, S.183]. Mit Asterisk für die Sitzungssteuerung und -vermittlung lässt sich dieses Problem umgehen, indem für den Anwendungsserver eine andere URI verwendet wird. Ein eingehender Anruf aktiviert in Asterisk dann eine Extension, die gleichzeitig das registrierte Endgerät des Nutzers anruft und die URI des IPTV-Call-Notifiers. Anschließend sendet der Anwendungsserver eine Nachricht an den Media-Server, so dass dieser in die aktuell wiedergegebenen Videodatenströme einen Hinweis einblenden kann.

Die Variante, dass ein SIP-fähiger Mediaplayer die Benachrichtigung selbst verarbei-

tet, lässt sich mit dem Programm *MPlayer*⁷ realisieren. Dieser Mediaplayer lässt sich als SIP-Client für eingehende Verbindungen nutzen. Dazu muss jedoch durch ein externes Skript eine REGISTER-Nachricht geschickt werden, damit der Proxy eine eingehende Verbindung auf die dauerhafte URI des Nutzers auch zum Mediaplayer weiterleitet. Der Player selber unterstützt die Registrierung nicht, da die SIP-Funktionalität nur zum Abhören von Voicemail-Nachrichten gedacht ist.

Ausblick

Mit der zunehmenden Verbreitung von IPTV-Set-Top-Boxen ist es vorstellbar, dass diese Set-Top-Boxen zusätzlich einen SIP-UA beinhalten. Dann wäre ein solcher Benachrichtigungsdienst nicht mehr nötig, da der Nutzer mit seiner Set-Top-Box und den angeschlossenen Aufnahme- und Wiedergabegeräten die eingehende Sitzung direkt annehmen kann.

7.3.6 Veranstaltungskalender

Der Veranstaltungskalenderdienst soll den Nutzer über aktuelle Ereignisse aus einem öffentlichen Kalender benachrichtigen. Diese Benachrichtigungen soll er auf seinem SIP-Endgerät per Textnachricht oder Sprache erhalten.

Realisierung

Die Grundlage für diesen Dienst bildet ein Wake-Up-Dienst, welchem der Nutzer über spezielle Textnachrichten eigene Erinnerungen hinterlegen kann.

Dieser Dienst kann durch die Nutzung einer zentralen Datenquelle erweitert werden. Ein sehr verbreitetes Format für öffentliche Kalender im Internet ist *iCalendar* [RFC5545]. Es existieren unzählige öffentlich verfügbare Kalender, in denen unterschiedlichste Ereignisse, wie Fußballspiele, Formel-1-Autorennen, Theaterspielpläne oder regelmäßige Treffen von Ortsverbänden, enthalten sind.

Diese Informationen können als Grundlage verwendet werden, um einen Veranstaltungskalender mit Benachrichtigungen über SIP zu erzeugen. Dazu können ausgewählte Kalender oder einzelne Elemente von einem *iCalendar-Resource-Adapter* in einen Wake-Up-Service übertragen werden. Dieser Service generiert dann zum vorgegebenen Zeitpunkt, also beispielsweise 30 Minuten vor dem Beginn eines Bundesligafußballspiels, eine SIP-Textnachricht und schickt sie an die URI des Nutzers. Falls sein aktuell genutzter UA nur Audiodatenströme unterstützt, kann über ein Sprachsynthesystem

⁷<http://www.mplayerhq.hu/>

auch eine Sprachnachricht generiert werden. In diesem Fall kann der Nutzer auch über ein sprachbasierendes Menü einen zweiten Rückruf aktivieren, z.B. zu Beginn des Ereignisses.

Über Anbieter für private Kalender mit iCalendar-Schnittstelle, wie etwas *Google Calendar*, kann der Nutzer nicht nur auf Veranstaltungskalender zugreifen, sondern kann die Funktionalität auch mit einem persönlichen Terminplaner kombinieren.

Ausblick

Generell kann ein solcher Veranstaltungskalender auch auf einer aktiven Abfrage des Nutzers basieren. D.h. der Kalender verschickt keine Nachrichten, sondern der Nutzer fragt den Kalender über eine SIP-Nachricht oder ein Sprachmenü ab und bekommt so eine Auswahl der für ihn in diesem Moment interessanten Veranstaltungen.

Da die Anwendungsserver für das NGN in den meisten Fällen auch eine Schnittstelle zu Abrechnungssystemen des Providers besitzen, kann der Veranstaltungskalender auch zu einem interaktiven Buchungssystem aufgewertet werden. Der Nutzer kann dann, zu der von ihm ausgewählten Veranstaltung, direkt Eintrittskarten erwerben und über sein Kundenkonto beim Provider bezahlen. Die Eintrittskarte kann ihm anschließend, vergleichbar mit dem Handyticket⁸ der Deutschen Bahn, per MMS auf sein mobiles Endgerät geschickt werden. Diese Tickets enthalten einen 2D-Matrixcode, der dann am Einlass zur Veranstaltung mit einem entsprechenden Lesegerät ausgelesen werden kann.

Eine interessante Erweiterung dieses Systems ist durch die Zuhilfenahme von kontextbezogenen Nutzerinformationen möglich. Über Bewertungen, die der Nutzer nach den Veranstaltungen abgibt und den Bewertungen der Veranstaltungen von Nutzern mit ähnlichen Profilen, lässt sich bei zukünftigen Abfragen die Auswahl an Veranstaltungen besser auf den Nutzer zuschneiden. Er würde dann mehr Veranstaltungen erhalten, die seinen Interessen entsprechen und durch den Vergleich mit ähnlichen Nutzerprofilen Hinweise zu interessanten Veranstaltungen erhalten, die er etwa über eine Kategoriesuche nicht gefunden hätte.

7.3.7 Netzbasierende Kontaktliste

Die netzbasierende Kontaktliste soll ein Telefonbuch sein, was mit verschiedenen SIP-Endgeräten genutzt werden kann, ohne dass die enthaltenen Adressen auf den Endgeräten gespeichert sind. Diese Funktion ist besonders im Bereich der persönlichen

⁸http://www.bahn.de/p/view/buchung/mobil/handy_ticket.shtml

Mobilität (siehe Kapitel 3.5.2.1) wichtig, da man sich zwar von verschiedenen UAs mit seiner persönlichen URI registrieren kann, die Telefonnummern und SIP-URIs seiner Gesprächspartner dort jedoch meist nicht zur Verfügung stehen.

Realisierung

Eine einfache Anwendung für diesen Zweck stellt Asterisk selbst bereit. Das VoiceMail-System in Asterisk beinhaltet eine *Dial-by-Name*-Funktion, bei welcher der Nutzer über die Eingabe der ersten drei Buchstaben eines gespeicherten Kontaktes mit dieser Extension verbunden wird [Wint09a, S.212 f.]. Diese Funktion beinhaltet jedoch keine Rückmeldung, so dass man sie blind bedienen muss und erst weiß welche Extension aufgerufen wurde, wenn auf der anderen Seite das Gespräch angemommen wird.

Eine effizientere Lösung ist durch die Verwendung des Sprachdialogsystems (IVR⁹) in Asterisk möglich. Damit lässt sich ein Sprachmenü für die Auswahl eines Kontaktes erzeugen. Der Nutzer gibt dann, wie bei *Dial-by-Name*, die ersten Buchstaben eines Kontaktes ein. Die Kontakte sind in diesem Fall in der Asterisk-Datenbank gespeichert. Wird ein passender Eintrag gefunden, wird eine Extension aufgerufen, die den Nutzer nicht sofort mit seinem Ziel verbindet, sondern vorher über ein Sprachsynthesystem den Namen des gewählten Gesprächspartners vorliest. Erst nach einer Bestätigung durch den Nutzer wird anschließend eine Verbindung aufgebaut.

Ausblick

Eine Realisierung der Kontaktliste in der Service- und Anwendungsschicht des NGN ermöglicht eine deutlich leistungsfähigere Lösung. Zum einen kann ein Anwendungsserver den Namen neben der Eingabe über die Telefontastatur auch durch eine Spracherkennungssoftware entgegennehmen.

Daneben sind aber auch neue Ansätze für die Bereitstellung des Telefonbuchs möglich. So kann das Telefonbuch auch per LDAP direkt auf die Endgeräte verteilt werden. So steht nach der Registrierung über die persönliche URI sofort die gesamte Kontaktliste des Nutzers auf dem gerade verwendeten Endgerät zur Verfügung. Dieses muss dafür aber den Datenaustausch per LDAP unterstützen. Dabei können auch öffentliche LDAP-Adressbücher, etwa das der TU Ilmenau eingebunden werden. Bei Clients für Instant-Messaging (ICQ, MSN etc.) ist eine zentral gespeicherte Kontaktliste schon länger üblich. Diese wird vom Server gleichzeitig mit Präsenzinformationen über den

⁹Interactive-Voice-Response-System

Online-Status der Gesprächspartner zum lokalen Client übermittelt. Eine solche Funktionalität ist für SIP-UAs ebenfalls denkbar.

7.4 Ausgangspunkt für weitere Dienste

7.4.1 Mitgelieferte Resource-Adapter

Die Mobicents-Plattform enthält bereits eine Reihe Komponenten, die zur Erzeugung von komplexen Diensten oder zur Kommunikation über bestimmte Protokolle genutzt werden können. Die Resource-Adapter sind dabei für die Kommunikation über ein bestimmtes Protokoll oder mit einem speziellen System zuständig.

Text-to-Speech

Der Text-to-Speech-Resource-Adapter basiert auf der FreeTTS-Sprachsynthese. Über diesen RA können beliebige JSLEE-Anwendungen auf die Sprachsynthese zurückgreifen. Die FreeTTS-Engine ist komplett in Java geschrieben und ist als Opensource verfügbar. Momentan existieren jedoch nur englischsprachige Bibliotheken, so dass für die Verwendung von deutscher Sprache auf ein anderes Sprachsynthesesystem zurückgegriffen werden muss. Die TTS-Engine erzeugt Audiodateien, welche anschließend über den Media-Player-RA per RTP in einer SIP-Sitzung abgespielt werden.

Media Server

Der Media Server der Mobicents-Plattform ist für die Aufnahme und Wiedergabe von Medienströmen über RTP gedacht. Er kann zusätzlich auch Konferenzfunktionen bereitstellen. Er wird über einen eigenen Media-Player-RA an Mobicents angebunden und kann so von allen JSLEE-Anwendungen genutzt werden.

Asterisk-RA

Der Asterisk-Resource-Adapter ermöglicht die Nutzung des Asterisk Management Interface (AMI, siehe Kapitel 6.3.1). Darüber können alle Nachrichten der Verbindungs- und Sitzungssteuerung abgefangen und ausgewertet werden. Diese Nachrichten können beispielsweise für Benachrichtigungsfunktionen (siehe IPTV-Call-Notifier 7.3.5) genutzt werden, ohne dass sich der AS als UA registrieren muss. Der Eingriff in die Kommunikation über AGI ist jedoch noch nicht möglich.

XMPP-RA

Mit dem XMPP-RA ist das *Extensible Messaging and Presence Protocol* (XMPP) nutzbar. XMPP ist ein Echtzeit-XML-Streaming Protokoll [RFC3920], welches beispielsweise für verschiedene IM-Anwendungen, wie Google Talk oder JABBER, genutzt wird. Es wird aber auch bei Systemen zur Echtzeit-Kollaboration, wie Google Wave¹⁰, angewendet. Darauf aufbauend existiert in den Beispielen zur JSLEE-Plattform ein Google-Talk-Bot, der automatisiert mit dem Nutzer kommunizieren kann. Dieser kann für vielfältige Zwecke weiterverwendet werden, etwa für Benachrichtigungen aus Kalendern, Börsennachrichten oder für Erinnerungsfunktionen.

weitere RAs

Im Lieferumfang der Mobicents JAIN SLEE befinden sich noch weitere RAs für folgende Protokolle:

- Diameter
- HTTP Client
- HTTP Servlet
- JAIN MGCP
- Parlay
- Rules (zum dynamischen Erzeugen von Sprachmenüs)
- SMPP (Short Message Peer to Peer)
- XCAP (XML-basierende Konfigurationsprotokoll)

7.4.2 Kombination von Funktionsblöcken

Die vorgestellten Dienstkonzepte setzen sich alle aus einzelnen Funktionsblöcken zusammen. Dabei wird deutlich, dass häufig mehrere Funktionsblöcke oder verschiedene Anwendungen genutzt werden können um den gleichen Dienst zu erzeugen (Abb. 7.12).

Einige Dienst konnte mit den Funktionen der Asterisk API umgesetzt werden, die jedoch hauptsächlich auf die Erzeugung von Routing-Dienstleistungen ausgelegt ist. Aufgrund der in Kapitel 6.3.3 beschriebenen Randbedingungen erscheint es sinnvoller, solche Dienstleistungen als SIP-Servlet zu implementieren. Diese bieten eine deutlich höhere Sicherheit, da kein direkter Zugriff auf die Steuerelemente benötigt wird, und bieten gleichzeitig eine vollwertige SIP-Schnittstelle.

¹⁰<http://www.waveprotocol.org/draft-protocol-specs/draft-protocol-spec>

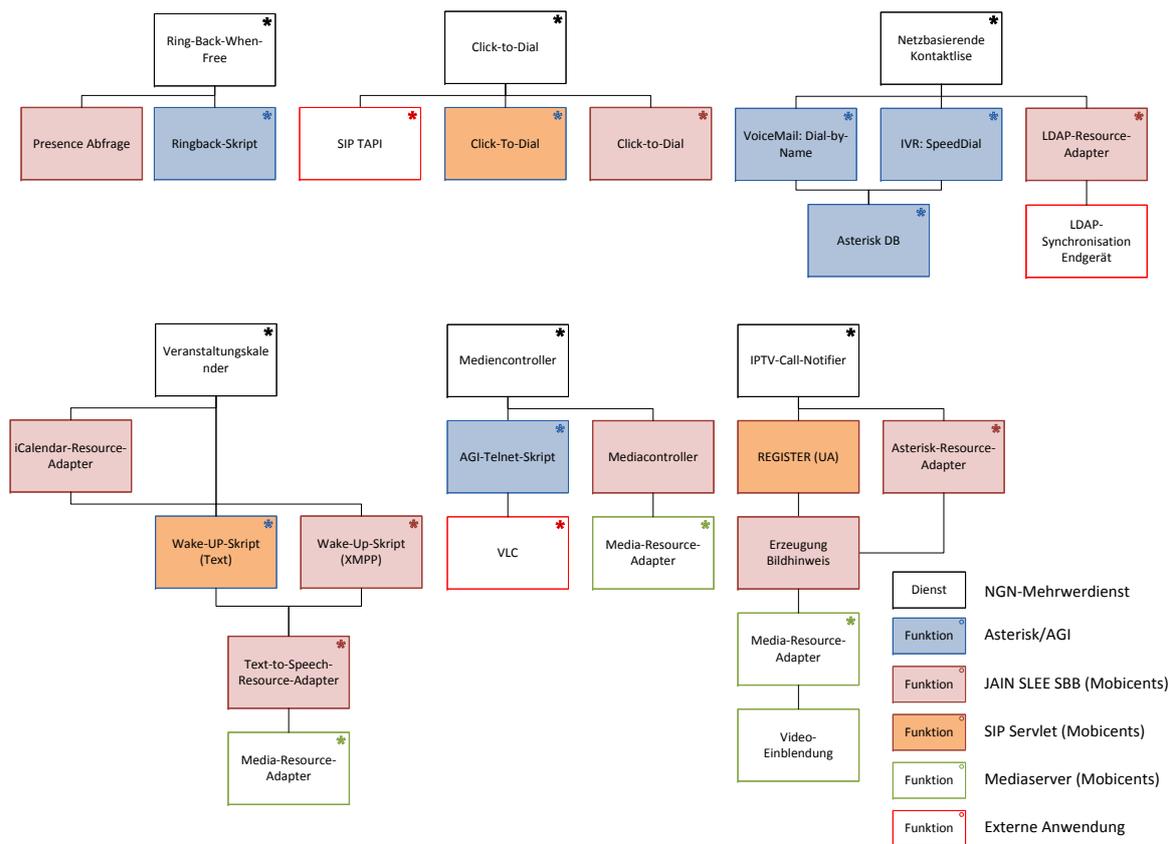


Abbildung 7.12: Übersicht über die Funktionalitäten der vorgestellten Dienstkonzepte, die Funktion der mit * gekennzeichneten Blöcke wurde entweder bereits implementiert oder lässt sich über mitgelieferte Softwarekomponenten nutzen.

Mit der großen Anzahl an mitgelieferten Resource-Adaptoren können auf Basis der JAIN-SLEE-Plattform viele der in Kapitel 4.3 beschriebenen Mehrwertdienste implementiert werden. Für Anwendungen, wie den iCalendar-Datenimport bietet sich die Entwicklung eigener Resource-Adapter an, so dass diese Funktion anschließend für andere Dienste ebenfalls zur Verfügung steht.

Neue Dienste können so auf die gesamte Funktion eines bestehenden Dienstes zurückgreifen oder nur Teile der Funktionalität nutzen. Durch die Erweiterung und die Kombination dieser Dienste können komplexe Mehrwertdienste erzeugt werden, die dann ein breiteres Feld an Funktionen abdecken als die umgesetzten einfachen Mehrwertdienste.

8 Fazit

8.1 Zusammenfassung

Das NGN bietet im Vergleich zu den herkömmlichen Telefonnetzen eine große Menge neuer Funktionen und Nutzungsmöglichkeiten. Sprachtelefonie ist dabei nur noch eine von vielen Dienstleistungen. Durch den modularen Aufbau des Netzes mit getrennten Schichten für Datentransport, Sitzungssteuerung und Anwendungserzeugung ist das Netz flexibel erweiterbar, da die Abhängigkeiten zwischen den Schichten gegenüber ISDN deutlich abgenommen haben.

Für die Netzbetreiber bietet sich durch die Vereinheitlichung der Steuerungs- und Verwaltungsfunktionen, sowie der Reduktion auf ein einziges Kernnetz, ein erhebliches Sparpotential gegenüber getrennten Netzen. Die Einführung eines IP-basierenden Kernnetzes bringt aber auch zahlreiche Probleme mit sich, die noch nicht vollständig gelöst sind. Die größten Probleme sind die Bereitstellung einer hohen Sprachqualität, die durch die Einführung von Quality-of-Service-Technologien gewährleistet werden soll, und die allgemeine Sicherheit in IP-Netzen, die eine gewisse Abschottung des Netzes gegenüber dem Internet nach sich ziehen wird.

Durch die ständig sinkenden Preise für die reine Sprachtelefonie sind die Netzbetreiber auf neue Verdienstmöglichkeiten durch Mehrwertdienste im NGN angewiesen. Für die Erzeugung von Mehrwertdiensten wurden im NGN mehrere Schnittstellen standardisiert, die zu großen Teilen auf bereits etablierten Technologien oder Anwendungen basieren und diese um eine Schnittstelle zum NGN erweitern. Dabei haben sich vordergründig zwei Java-basierende Spezifikationen für Anwendungsserver durchgesetzt. Einerseits eignen sich die SIP-Servlets besonders für einfache, plattformunabhängige und schnell zu implementierende Dienste. Andererseits steht mit der, auf der Java Enterprise Environment basierenden, JAIN-SLEE-Plattform eine Umgebung für komplexe Dienste zur Verfügung, die auch auf Medien- und Präsenzserver zurückgreifen kann. Für beide Plattformen existieren bereits Implementierungen, die als freie und kommerzielle Software vertrieben werden. Bei der Untersuchung dieser Plattformen hat sich die freie Software Mobicents als eine vielseitige und modulare Basis für die Bereitstel-

lung von Diensten im NGN erwiesen. Sie ermöglicht die Umsetzung von Diensten in einem Anwendungsserver, der sowohl SIP-Servlets als auch JAIN-SLEE-Anwendung unterstützt, und beinhaltet zusätzlich einen Medien- und einen Präsenzserver. Über Resource Adapter können beliebige Protokolle oder Systeme in die Diensterzeugung einbezogen werden.

Auf Grundlage dieser Plattform wurde ein Konzept für NGN zur Erprobung von Mehrwerdiensten entworfen und umgesetzt. Den Kern dieses Netzes bildet die IP-Telefonanlage Asterisk, die alle Sitzungs- und Verbindungssteuerungsfunktionen bereitstellt. Das aufgebaute NGN kann über verschiedene Zugangsnetze genutzt werden und ermöglicht Gespräche in andere Kernnetze, wie ISDN oder GSM. Mit Skype wurde zudem ein bereits stark verbreitetes VoIP-Netz nutzbar gemacht.

Anhand verschiedener prototypisch umgesetzter Dienste und den vorgestellten Dienstkonzepten wurde gezeigt, dass für die Erzeugung von Mehrwertdiensten verschiedene Plattformen und Ansätze zum Einsatz kommen können. Je nach Art der Dienstleistung können diese Dienste innerhalb eines Anwendungsservers oder über direkt über Schnittstellen von Asterisk realisiert werden.

8.2 Ausblick

Die vorgestellten Dienstkonzepte bieten einen Ausblick auf mögliche Mehrwertdienste, wie sie in absehbarer Zeit für alle Nutzer im NGN zum Einsatz kommen können. Die grundlegenden Dienstleistungen des NGN werden heute bereits teilweise von Anbietern für IP-basierende Telefonanschlüsse oder IPTV genutzt. Darüber hinaus sind bisher nur wenige innovative Mehrwertdienste für den Endnutzer verfügbar. Weil durch das Netzkonzept des NGN die Bereitstellung solcher Mehrwertdienste nicht mehr nur auf die Betreiber der Kernnetze beschränkt ist, werden mit hoher Wahrscheinlichkeit viele der vorgestellten Dienste von anderen Anbietern in naher Zukunft umgesetzt werden.

Für einige interessante Dienste, wie den Device-unify-Service, der mehrere Endgeräte zu einem Kommunikationszugang vereinheitlichen soll, bestehen noch technische Hürden, so dass eine Umsetzung einen interessanten Ausgangspunkt für weitere Forschung zur Realisierung dieser Konzepte bildet. Von besonderem Interesse werden in Zukunft auch sogenannte komplexe Dienste sein, welche die Funktionalitäten anderer, einfacher Dienste zu einer neuen Funktion zusammenfassen. Besonders die vorgestellte JAIN-SLEE-Plattform bietet mit ihren Service-Building-Blocks (SBB) gute Voraussetzungen für eine Untersuchung der Kombination von bestehenden Diensten, wobei auch auf die hier implementierten Dienstkonzepte zurückgegriffen werden kann.

Anhang

A Software und Konfigurationsdateien

A.1 Einrichtung Asterisk

Die Einrichtung der lokalen Testsysteme mit Asterisk wurde nach der Anleitung in [Wint09a, S.337 ff.] durchgeführt. Das Zielsystem, auf dem der Anwendungsserver anschließend installiert wird, ist bereits nach den Spezifikationen in [StMa08, GüRu08] vorbereitet und wird nur um den Anwendungsserver und das SIP-Gateway erweitert. Wie in Kapitel 7.1.1 beschrieben, werden die Konfigurationen im Dialplan über AEL durchgeführt.

A.1.1 Anbindung an Sipgate

Um über das PSTN-Gateway des Providers Sipgate.de Anrufe tätigen und entgegennehmen zu können, muss sich Asterisk dort als UA registrieren. Gleichzeitig muss im Fall einer NAT-Firewall zwischen Asterisk und dem Internet die dynamische IP-Adresse bekannt sein. Über einen DynDNS-Provider lässt sich einer dynamischen IP-Adresse ein fester DNS-Eintrag zuweisen, über welchen Asterisk selbstständig die eigene IP-Adresse herausfinden kann.

Listing A.1: Einrichtung des PSTN-Gateways über Sipgate (sip.conf)

[general]

```
port=5060
bindaddr=0.0.0.0
qualify=no
disallow=all
allow=alaw
allow=ulaw
allow=g729
allow=gsm
allow=slinear
```

```

srvlookup=yes
register => 9840217:EPZUCV@sipgate.de/9840217
nat=yes
externhost=XXXX.dynalias.org
externrefresh=10
localnet=10.0.0.0/255.0.0.0
(...)
[sipgate-out]
type=friend
insecure=invite
nat=yes
username=984XXXX
fromuser=984XXXX
fromdomain=sipgate.de
secret=XXXXX
host=sipgate.de
qualify=yes
canreinvite=no
dtmfmode=rfc2833
context = from-sipgate

```

Die eingehenden Gespräche sind damit dem Context `from-sipgate` und die ausgehenden Gespräche dem Context `sipgate-out` zugewiesen. Um ausgehende Gespräche über Sipgate zu leiten, müssen alle Nummern die nicht mit einem Pattern für interne Nummern gefunden werden, über die Applikation

`Dial(SIP/${EXTEN}@sipgate-out,120,trg);` weitergeleitet werden. Da bei der Verwendung von internen Nummerierungen die Extensions und damit die dazugehörigen Pattern deutlich kürzer sind, wird Asterisk bei der Auswahl einer internen Nummer (etwa 2001) immer zuerst diese aufrufen, bevor es den Anruf über Sipgate weiterleitet [Wint09a, S.67]. Die eingehenden Gespräche werden in dieser Konfiguration immer fest an eine bestimmte interne Nummer weitergeleitet.

Listing A.2: Einrichtung des PSTN-Gateways über Sipgate (`extensions.ael`)

```

context meine-telefone {
    _X. => {
        Set(CALLERID(num)=9840217);
        Dial(SIP/${EXTEN}@sipgate-out,120,trg);
        Hangup();
    }

```

```
}  
context from-sipgate{  
    9840217 => {  
        Dial(SIP/2001);  
    }  
}
```

Eine Ausführliche Anleitung findet sich in [Wint09a] und in der ständig aktualisierten Onlineausgabe unter [Wint09b].

A.1.2 Anbindung an Skype (SipToSis)

SipToSis verbindet als SIP-UA über die Skype-API das Skype-Netzwerk mit einem SIP-basierenden Netz (siehe Kapitel 6.3.2). SipToSis muss auf dem Computersystem laufen, auf welchem der Skype-Client ausgeführt wird [MHSp09]. Zur Konfiguration sind prinzipiell zwei Wege möglich. SipToSis kann ohne irgendwelche weitere Anmeldungen die eingehenden Gespräche an eine feste URI weiterleiten. Im Fall eines verwalteten Netzes ist die Einrichtung der Registrierung sinnvoll, da jeder Nutzer SipToSis so über eine feste URI erreichen kann, und Asterisk etwa bei Belegung des Kanals den Rückrufdienst nutzen kann.

Die Einrichtung ohne das PSTN-Gateway von Skype erfolgt über drei Dateien. In der Datei `siptosis.cfg` werden die Verbindungsparameter und die Registrierungsdaten festgelegt. In den Dateien `SkypeToSipAuth.props` und `SipToSkypeAuth.props` werden die Regeln für die Weiterleitung eingehenden und ausgehender Gespräche im Skype-Netz festgelegt.

Listing A.3: Ausschnitt der Datei `siptosis.cfg` für die Anmeldung bei Asterisk

```
host_port=5070  
contact_url=sip:2002@10.0.0.1:5060  
from_url="2002" <sip:2002@10.0.0.2:5060>  
username=2002  
realm=asterisk  
passwd=1234  
expires=3600  
do_register=yes  
minregrenewtime=120  
regfailretrytime=15
```

Eingehende Skypeverbindungen können entweder alle über den regulären Ausdruck `*` abgefangen werden und zu einer URI weitergeleitet werden oder über den Skype-Namen gefiltert werden. Es können auch Skypekontakte blockiert werden

Listing A.4: `SkypeToSipAuth.props` (eingehende Skype-Verbindungen)

```
skypeuser1,deny
sykepuser2,sip:984XXX@sipgate.de:5060
*           ,sip:2001@10.4.1.105:5060
```

Für ausgehende Verbindungen in das Skypenetz besteht die Schwierigkeit, dass die meisten SIP-Endgeräte nur numerische URIs aufrufen können. Dafür existiert in der Datei `SipToSkypeAuth.props` ein Dialplan, über welchen die Skypenamen von Kontakten verschiedenen einfach einer Durchwahlnummer zugeordnet werden. Diese Liste muss jedoch entweder manuell angepasst werden oder über eine Webanwendung so bereitgestellt werden, dass andere Nutzer auf die Konfiguration zugreifen können.

Listing A.5: `SipToSkypeAuth.props` (ausgehende Skype-Verbindungen)

```
15615555555,*,*,skypeuser1
15554321555,*,*,echo123
15615555554,*,172.17.*.*,skypeuser1
```

A.2 Einrichtung Mobicents

Mobicents kann wie jede Openourceanwendung auf mehreren Wegen installiert werden. Die einfachste und schnellste Variante ist die Installation als Binärpaket. Weiterhin kann Mobicents auch als Quellcode aus dem SVN des Mobicents-Projekt ausgecheckt werden und anschließend selbst übersetzt werden (siehe A.3.2).

Bei der Installation als Binärpaket muss der Nutzer entscheiden ob er nur den Mobicents JAIN SLEE-Server braucht, oder ob er den Media- und Presenceserver sowie die SIP-Servlets ebenfalls braucht. Ebenso sind Pakete mit und ohne integrierten JBoss-Server verfügbar. Im Quellcode muss jeder Server getrennt ausgecheckt und installiert werden.

Das Setup des Mobicents Anwendungsservers als Binärpaket wird in [MSIR⁺09] und [Sila08] beschrieben. Da Mobicents eine JAVA-Anwendung ist, ist es gleichermaßen unter Windows und Linux lauffähig.

Im Oktober 2009 waren Version 1.2.1 des Gesamtpaketes und Version 1.2.3 des JAIN SLEE aktuell. Obwohl schon von JBoss bereits Version 5.1 verfügbar war, basierte der

Mobicentsserver noch auf Version 4.2.3 des JBoss Microcontainers.

A.2.1 Installation des Java JDK/JRE

Mobicents bzw. JBoss benötigen zwingend die Sun Java 5.0 Runtime Environment (JRE), alternative (freie) Java-Versionen funktionieren nicht. Zur Entwicklung von Anwendungen ist jedoch eher das Java Development Kit (JDK) ratsam, da dort die Dokumentation der JAVA-API beinhaltet ist. Für die Entwicklung von SIP-Servlets kann auch die JavaEE¹ verwendet werden, da dort bereits das Java WebToolkit enthalten ist. Auf einer 64-bit Plattform empfiehlt sich die Installation der 64-bit Java-VM, da so ein mehr Speicher nutzbar ist (sonst 1,5 GB). Wird nicht soviel Speicher benötigt, profitieren Anwendungen nicht weiter davon und werden nicht schneller laufen als die 32-bit VM [Sila08, 2.1].

Unter Debian/Ubuntu kann das Java-JDK auch direkt über den Befehl

```
apt-get install sun-java5-jdk
```

installiert werden.

A.2.2 JAVA-Umgebungsvariable

In [Sila08, 2.2] wird beschrieben, dass die Umgebungsvariable `JAVA_HOME` optional ist. In den praktischen Versuchen hat sich das Fehlen der Variable als Fehlerquelle dargestellt. Es empfiehlt sich also sie trotzdem entsprechend anzupassen.

Unter Linux sollte der Befehl `echo $JAVA_HOME` etwa einen solchen Output produzieren:

```
# echo $JAVA_HOME
/usr/lib/jvm/java-1.5.0-sun
```

A.2.3 Auswahl der JVM

Bei den meisten Linux-Distributionen wird heute eine freie Java-VM mitgeliefert. Bevor Mobicents ausgeführt wird, muss erst die Sun JVM als Systemstandard festgelegt werden.

```
update-alternatives --config java \\  
...
```

¹<http://java.sun.com/javase/downloads/index.jsp>

A.2.4 Paketauswahl und Download

Für die Installation stehen mehrere Binärpakete bereit. Empfehlenswert ist die Verwendung eines Komplettpaketes, was die JAIN-SLEE, den SIP-Servlet-Container, den Media- und den Presence Server zusammen mit dem JBoss-Anwendungsserver beinhaltet. Diese Pakete sind unter ² zu finden. Für den SIP-Servlet-Container und die JAIN SLEE sind auch einzelne Pakete mit und ohne JBoss verfügbar.

Diese Pakete werden anschließend in den Zielordner extrahiert.

A.2.5 Konfiguration

Anschließend sollte auch die `JBOSS_HOME`-Variable angepasst werden. JBoss kann unter Windows oder Linux auch selbstständig versuchen zu erkennen aus welchem Verzeichnis heraus es gestartet wurde, was allerdings nicht immer zuverlässig funktioniert. Die Variable sollte hinterher in das Verzeichnis des JBoss-Servers zeigen, etwa

```
JBOSS_HOME="/tmp/mobicents-all-1.2.0.GA-jboss-4.2.3.GA/jboss-4.2.3.GA"
```

A.2.6 Starten

Gestartet werden kann der JBoss-Anwendungsserver über die `run`-Skripte im Verzeichnis `jboss-4.2.3.GA/bin`. Dabei sollte darauf geachtet werden, dass der Anwendungsserver sich an die korrekte IP bindet. Das kann über den Parameter `-b` auch manuell geschehen.

```
/tmp/.../jboss-4.2.3.GA/# ./run.sh -b 0.0.0.0
```

Bei Start können folgende Fehler auftreten:

- Ein Resource-Adapter kann keine Verbindung herstellen (etwa zu AMI)
- ein SIP-Client läuft bereits lokal und blockiert den Port (z.B. 5060)
- die Variable `JBOSS_HOME` wurde ohne Schrägsstrich abgeschlossen

Zum Anhalten können die `Shutdown`-Skripte im gleichen Ordner verwendet werden (siehe A.3.1.3).

²<http://www.mobicents.org/jainsleeplatformdownload.html>

A.3 Setup der Entwicklungsumgebung

In diesem Abschnitt wird die Einrichtung einer Entwicklungsumgebung zur Nutzung der Quellcode-Version von Mobicents erläutert. Eine kurze Anleitung ist unter [Mart08] zu finden. Zur Einrichtung der Entwicklungsumgebung für die Verwendung von SIP-Servlets steht unter [Tour09] eine Anleitung zur Verfügung.

A.3.1 Installation benötigter Tools

A.3.1.1 Apache Ant

Apache Ant ist ein Java-Programm zum automatisierten Erzeugen (Compilieren) von Programmen, welche im Quelltext vorliegen. Es ist vergleichbar mit dem bekannten Programm *make*.

Ant wird bei mobicents verwendet um Resource Adapter (RA) oder sogenannte Deployable Units (DU) automatisiert bereitzustellen, und ggf. Abhängigkeiten aufzulösen. Beispielsweise durch Installation benötigter RAs für den gerade installierten Dienst.

Das Programm kann von <http://ant.apache.org> bezogen werden, bzw. wird in den Debian/Ubuntu Paketquellen als *ant* angeboten. Es wird mindestens die Version 1.7 benötigt.

Zur Installation als Archiv von der Projektwebsite muss manuell der Pfad zum Programmverzeichnis in die Umgebungsvariable `ANT_HOME` eingefügt werden. Soll es in jedem Verzeichnis direkt über den Befehl `ant` aufrufbar sein, muss natürlich auch das Verzeichnis mit den Binären Paketen in der `Path`-Variable bekannt gegeben werden.

Eine erfolgreiche Installation sollte den Befehl **ant** in einem Verzeichnis ohne `build.xml` folgendermaßen quittieren:

```
# ant
Buildfile: build.xml does not exist!
Build failed
```

A.3.1.2 Apache Maven

Apache Maven ist ein Programm zur Vereinfachung des Erstellungsprozesses und ermöglicht die standardisierte Kompilierung, Test und Installation von Java-Programmen.

Die Konfiguration erfolgt dabei über eine XML-Datei mit dem Namen *pom.xml* (Project Object Model). Sie enthält Informationen über den Quelltext, verlinkte Bibliotheken und automatisierte JUnit-Tests.

Maven kann direkt von der Projektwebseite <http://maven.apache.org/> oder von den Debian/Ubuntu Paketquellen als *maven2* bezogen werden.

Nach der manuellen Installation aus einem Archiv muss der Pfad der binären Installationsdateien der Path-Variable hinzugefügt werden.

Nach einer erfolgreichen Installation sollte der Befehl **mvn** in einem leeren Verzeichnis folgendes Resultat liefern:

```
# mvn
[INFO] Scanning for projects...
[INFO] -----
[ERROR] BUILD FAILURE
[INFO] -----
[INFO]
[INFO] (...)
[INFO] -----
[INFO] For more information, run Maven with the -e switch
[INFO] -----
[INFO] Total time: < 1 second
[INFO] Finished at: Wed Oct 28 16:35:52 CET 2009
[INFO] Final Memory: 2M/61M
[INFO] -----
```

A.3.1.3 JBoss Anwendungsserver

Da die Quellcode-Pakete von Mobicents im Gegensatz zu den binären Paketen keinen JBoss-Anwendungsserver enthalten, muss dieser separat installiert werden.

Dazu muss von <http://www.jboss.org/jbossas/> das entsprechende Paket bezogen werden. Es wird empfohlen die Version *4.2.3.GA* zu nutzen, welche den binären Paketen von mobicents ebenfalls beiliegt.

In den Paketquellen zu Ubuntu wird zwar das Paket *jbossas4* angeboten, dieses enthält jedoch verschiedene Fehler.

Nach der Installation des Archives muss die `JBOSS_HOME`-Variable auf das Installationsverzeichnis (nicht auf die binären Dateien) zeigen.

Anschließend kann der Anwendungsserver über die Startskripte im Unterverzeichnis *bin* gestartet werden. Unter Windows wird dafür die Datei *run.bat* und unter Linux *run.sh* verwendet.

```
/home/bob/jboss-4.2.3.GA/bin # ./run.sh
```

```
JBoss Bootstrap Environment
JBOSS_HOME: /home/bob/jboss-4.2.3.GA
JAVA: /usr/lib/jvm/java-1.5.0-sun
(...)
16:56:11,965 INFO [Server] Starting JBoss (MX MicroKernel)...
16:56:11,967 INFO [Server] Release ID: JBoss [Trinity] 4.2.3.
GA (build: SVNTag=JBoss_4_2_3_GA date=200807181417)
(...)
16:56:26,120 INFO [Server] JBoss (MX MicroKernel) [4.2.3.GA
(build: SVNTag=JBoss_4_2_3_GA date=200807181417)]
Started in 14s:148ms
```

Anhalten kann man den Server indem man das *Shutdown*-Skript im gleichen Ordner mit dem Parameter *-S* aufruft.

```
/home/bob/jboss-4.2.3.GA/bin # ./shutdown.sh -S
Shutdown message has been posted to the server.
Server shutdown may take a while - check logfiles for completion
```

A.3.1.4 Entwicklungsumgebung Eclipse

Bei der Entwicklungsumgebung (IDE) können alle Java-kompatiblen Anwendungen genutzt werden, es wird aber Eclipse empfohlen. Für die Verwendung mit SIP-Servlets wird die JavaEE-Version von Eclipse³ empfohlen [Tour09]. Da diese Version die Fernverwaltung und die Steuerung des JBoss-Anwendungsservers unterstützt, ist sie auch für die Verwendung mit der JSLEE gut geeignet.

Für die Eclipse-IDE stehen zahlreiche Plugins zur Verfügung, beispielsweise zur Entwicklung von JAIN SLEE Anwendungen (siehe A.3.4) oder zur Nutzung von Apache Maven.

A.3.1.5 SVN-Client

Um den Quellcode vom mobicents-Projekt zu erhalten wird ein SVN-Client benötigt. Unter Linux kann einfach auf Subversion zurückgegriffen werden, was bei den meisten Distributionen unter dem Paketnamen *subversion* erhältlich ist. Unter Windows kann beispielsweise auf TortoiseSVN zurückgegriffen werden.

Es können auch Plugins für die Eclipse-IDE, wie Subclipse, genutzt werden.

³<http://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/galileo/SR1/eclipse-jee-galileo-SR1-win32.zip>

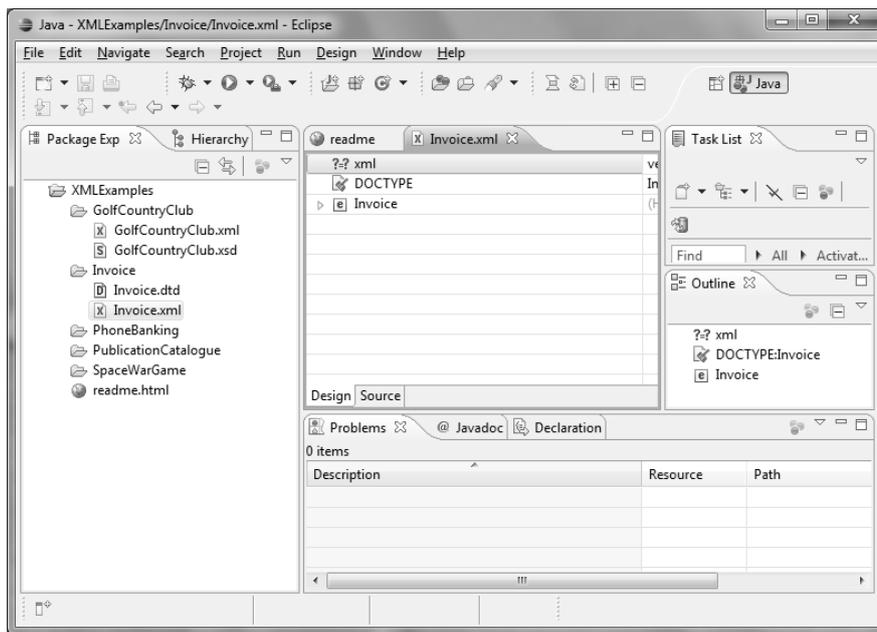


Abbildung A.1: Eclipse IDE für Java (hier unter Windows)

A.3.2 Installation Mobicents aus Quelltext

Jeder Server aus dem mobicents-Projekt muss einzeln installiert und übersetzt werden. Man kann jedoch alle Server zusammen per SVN auschecken. Die öffentliche URL für das SVN ist `http://mobicents.googlecode.com/svn/trunk/`. Da der Quellcode in diesem Verzeichnis aber aus der aktuellen Entwicklung stammt, und dementsprechend auch instabil sein kann, sollte der Branch der Version 1.x verwendet werden, beispielsweise für den JAIN SLEE Server `http://mobicents.googlecode.com/svn/branches/servers/jain-slee/1.x.y`.

Denn Quellcode aller Server erhält man per Checkout der entsprechenden Quellen:

```
/home/bob/workspace/servers # svn co http://mobicents.googlecode.com/svn/branches/servers
```

Um die einzelnen Server nun zu übersetzen und auf dem JBoss-Server zu installieren, führt man jeweils in den Verzeichnissen der Server den Befehl `mvn install` aus.

```
workspace/servers/jain-slee # mvn install
[INFO] Scanning for projects...
Downloading: http://repository.jboss.org/maven2/org/mobicents/mobicents/12/mobicents-12.pom
21K downloaded (mobicents-12.pom)
[INFO] Reactor build order:
[INFO] Mobicents :: JAIN SLEE Server
```

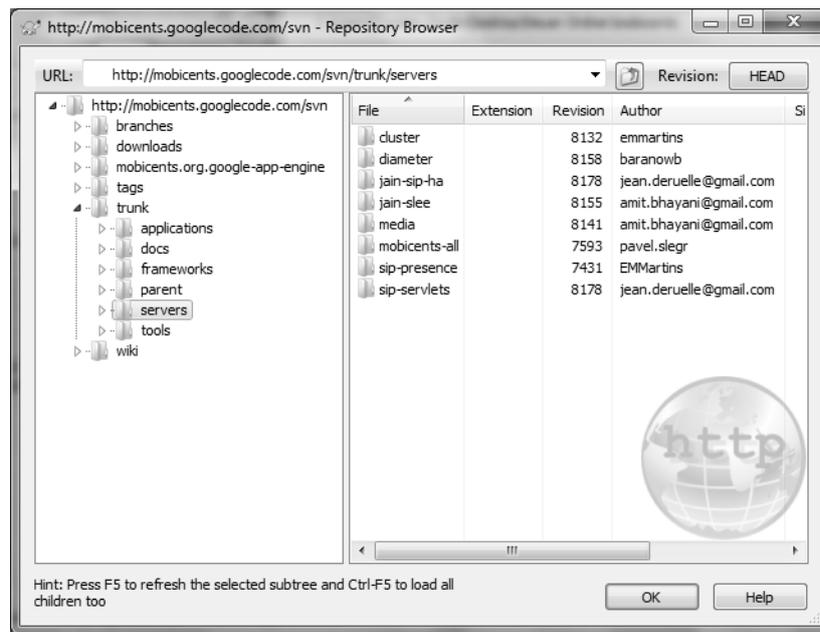


Abbildung A.2: Trunk des mobicents-Projekt in Repository-Browser von TortoiseSVN

```
(...)
```

```
[INFO] -----
```

```
[INFO] -----
```

```
[INFO] BUILD SUCCESSFUL
```

```
[INFO] -----
```

```
[INFO] Total time: 13 minutes 59 seconds
```

```
[INFO] Finished at: Wed Oct 28 18:27:01 CET 2009
```

```
[INFO] Final Memory: 81M/177M
```

```
[INFO] -----
```

Bei der ersten Ausführung versucht Maven alle benötigten Abhängigkeiten aus dem Repository in das lokale Repository herunterzuladen, was unabhängig von der Verbindungsgeschwindigkeit mehrere Minuten dauern kann.

A.3.3 Einrichtung Eclipse

Um das Projekt in Eclipse nutzen zu können, muss für Maven der Pfad zum Workspace in welchem das Projekt liegt in der Variable `M2_HOME` hinterlegt werden. Dies erfolgt mit dem Befehl:

```
/home/bob # mvn -Declipse.workspace=workspace eclipse:add-maven-repo
```

```
[INFO] Scanning for projects...
```

```
[INFO] Searching repository for plugin with prefix: 'eclipse'.
[INFO] org.apache.maven.plugins: checking for updates from
central
[INFO] org.codehaus.mojo: checking for updates from central
[INFO] artifact org.apache.maven.plugins:maven-eclipse-plugin:
checking for updates from central
Downloading: http://repo1.maven.org/maven2/org/apache/maven/pl
ugins/maven-eclipse-plugin/2.7/maven-eclipse-plugin-2.7.pom
(...)
[INFO] [eclipse:add-maven-repo {execution: default-cli}]
[INFO] -----
[INFO] BUILD SUCCESSFUL
[INFO] -----
[INFO] Total time: 2 minutes 23 seconds
[INFO] Finished at: Wed Oct 28 19:21:58 CET 2009
[INFO] Final Memory: 13M/111M
[INFO] -----
```

Dabei ist auch zu erkennen, dass Maven automatisch das *Maven-eclipse-plugin* herunterlädt und installiert.

Anschließend kann jeder Server einzeln als Projekt über die Funktion *General > Existing Projects into Workspace* in Eclipse importiert werden. Dazu gibt man das oberste Verzeichnis des jeweiligen Servers (z.B. `../workspace/jain-slee`) an. Dabei ist darauf zu achten, dass die Option zum Kopieren der Dateien in den Workspace deaktiviert ist.

Da noch die verlinkten Bibliotheken fehlen, müssen diese anschließend für jedes Projekt ins lokale Repository von Maven nachgeladen werden.

```
../workspace/jain-slee/1.x.y # mvn mobicents:eclipse
(...)
[INFO] -----
[INFO] BUILD SUCCESSFUL
[INFO] -----
[INFO] Total time: 11 minutes 7 seconds
[INFO] Finished at: Wed Oct 28 20:24:21 CET 2009
[INFO] Final Memory: 55M/385M
[INFO] -----
```

A.3.4 Eclipse-Plugins

Plugins für Eclipse werden normalerweise direkt aus Eclipse über die Funktion *Help > Install Software* installiert.

m2eclipse Zur Installation des Maven-Plugins für Eclipse, verwendet man als Quelle <http://m2eclipse.sonatype.org/update/>. Anschließend können die benötigten Komponenten ausgewählt und installiert werden.

Um für das bestehende Projekt das Maven-Plugin zu aktivieren muss nach dem Neustart von Eclipse über *Maven > Enable Dependency Management* die Verwaltung durch Maven bei dem Projekt aktiviert werden.

Mobicents-Plugins Für die Nutzung mit Mobicents stehen mehrere Plugins zur Verfügung. Die Plugins werden in Eclipse von der Adresse <http://mobicents.googlecode.com/svn/downloads/sip-servlets-eclipse-update-site/> installiert. Nach Bedarf können ein Plugin für den JAIN SLEE (EclipsLEE) und für die SIP Servlets (Sip Servlets Core Plugin) installiert werden.

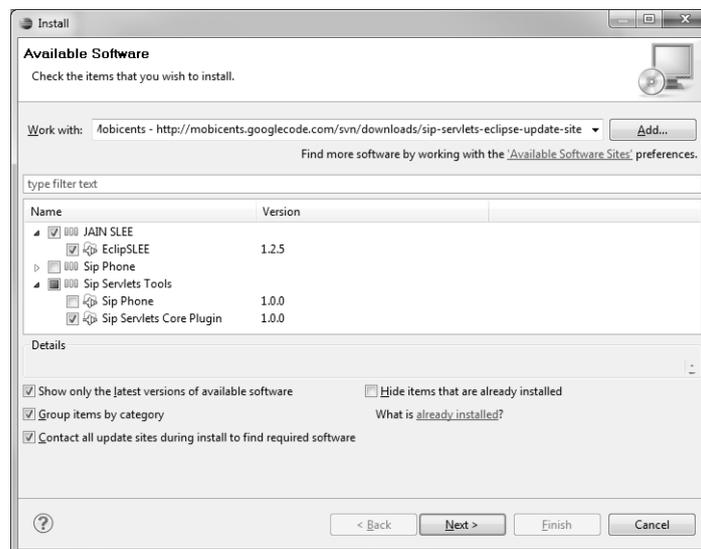


Abbildung A.3: Mobicents Plugins für Eclipse

A.3.5 Anlegen von Beispielen

SIP Servlet Die Einrichtung eines Beispiels für ein SIP-Servlet ist in [Tour09] beschrieben. Dazu muss der Entwicklungsumgebung Eclipse in der Server-Ansicht mitgeteilt werden was für ein Anwendungsserver zum Einsatz kommt (z.B. JBoss 4.2.3) und

wo sich das Wurzelverzeichnis des Anwendungsservers befindet. Anschließend kann der Anwendungsserver über den Server-Tab in Eclipse ferngesteuert werden.

Eine leere SIP-Servlet-Anwendung kann man nun erzeugen, indem im Menü *File* → *New* → *Dynamic Web Project* ausgewählt wird. Dabei muss allerdings die Version des Web-Moduls auf „2.4“ geändert werden und die Konfiguration „Converged SIP/Web Project“ ausgewählt sein.

JAIN SLEE SBB Für die Erzeugung eines leeren Projektes für die JSLEE ist eine Anleitung unter ⁴ zu finden. Die Erzeugung des Projektes erfolgt über eine Maven-Vorlage, die vorher im Verzeichnis `.m2` des Home-Ordners als `archetype-catalog.xml` definiert werden muss ⁵.

Listing A.6: Datei `archetype-catalog.xml`

```
<?xml version="1.0" encoding="UTF-8"?>
<archetype-catalog>
  <archetypes>
    <archetype>
      <groupId>org.mobicients.tools.maven.archetype.slee</groupId>
      <artifactId>jain-slee-basic-service</artifactId>
      <version>1.0.0.BETA1</version>
      <repository>http://repository.jboss.org/maven2</repository>
    </archetype>
    <archetype>
      <groupId>org.mobicients.tools.maven.archetype.slee</groupId>
      <artifactId>jain-slee-basic-ra</artifactId>
      <version>1.0.0.BETA1</version>
      <repository>http://repository.jboss.org/maven2</repository>
    </archetype>
  </archetypes>
</archetype-catalog>
```

Anschließend kann über den Befehl

```
# mvn -Declipse.workspace=. eclipse:add-maven-repo
```

ein Konfigurationsdialog zum Anlegen eines neuen JAIN-SLEE-SBB oder eines JAIN-SLEE-RA aufgerufen werden.

⁴http://www.jboss.org/feeds/post/p_a_jain_slee_beginner_s_tutorial_that_s_a_hellip

⁵<http://groups.google.com/group/mobicients-public/web/jainslee-maven-archetypes>

B Abbildungen

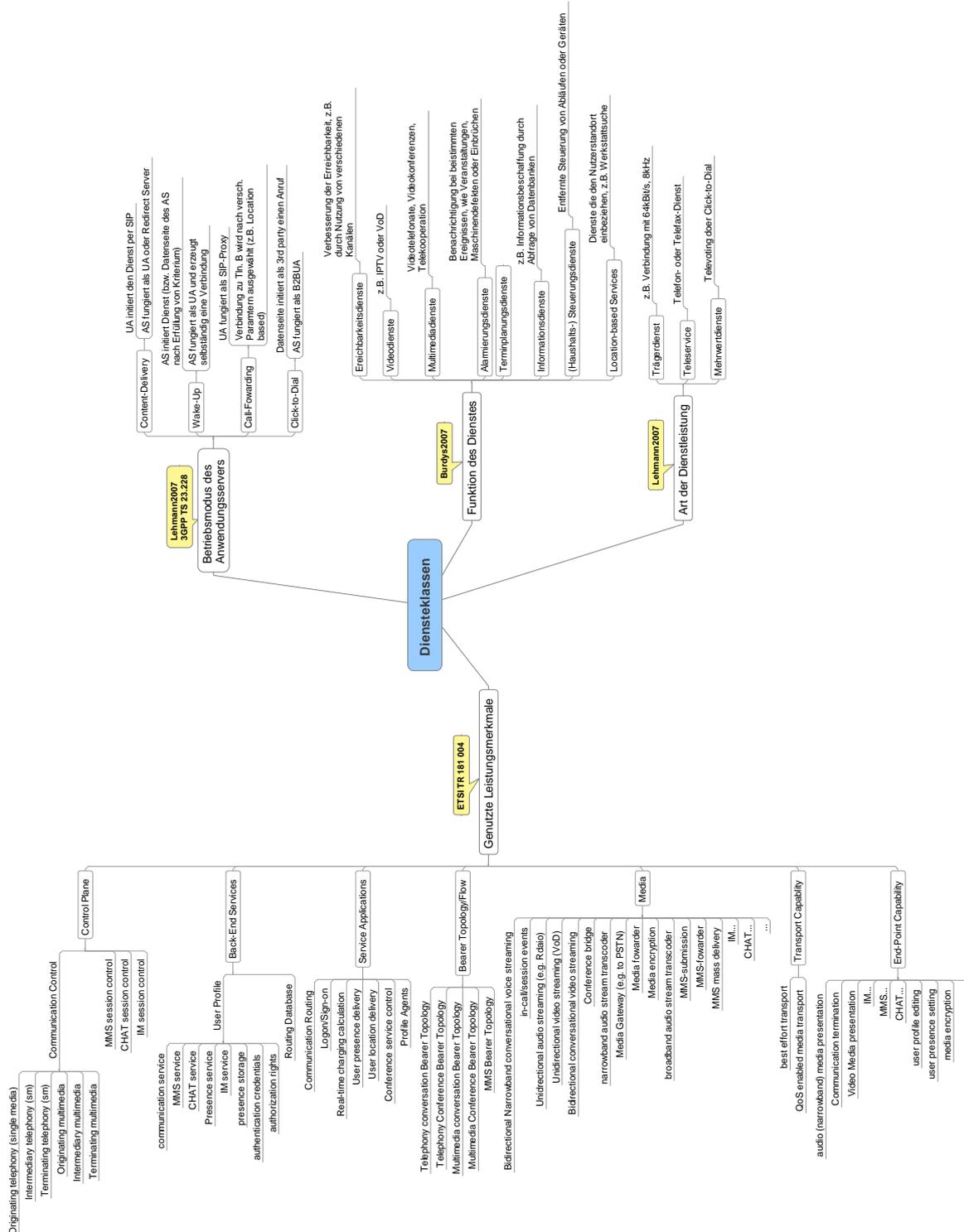


Abbildung B.1: Klassifizierungen für Dienste im NGN (detailliert)

Literaturverzeichnis

- [Bach09] Daniel Bachfeld. *Erster iPhone-Wurm unterwegs*. heise Security, <http://www.heise.de/security/meldung/Erster-iPhone-Wurm-unterwegs-853372.html>, November 2009. (Abgerufen am 20.11.2009).
- [BBDU00] Simon Beddus, Gary Bruce, Steve Davis und Steve Davis Ulticom. Opening Up Networks with JAIN Parlay. *IEEE Communications Magazine* 38(4), April 2000, S. 136–143.
- [Berg06] Stefan Berg. *Das IP Multimedia-Subsystem - Enabler für ein neues Dienstzeitalter?* DETECON Consulting, http://www.detecon-dmr.com/de/themes/technology/detail.html?unique_id=192794, April 2006. (Abgerufen am 04.09.2009).
- [Brie09a] Volker Briegleb. *Kabel Deutschland: Internet künftig mit 100 MBit/s*. heise online, <http://www.heise.de/newsticker/meldung/Kabel-Deutschland-Internet-kuenftig-mit-100-MBit-s-798025.html>, September 2009. (Abgerufen am 23.10.2009).
- [Brie09b] Volker Briegleb. *Telekom kündigt IPTV ohne Internetanschluss an*. heise online, <http://www.heise.de/newsticker/meldung/144848>, September 2009.
- [BuTr07] Sven Burdys und Ulrich Trick. Mobile Dienstplattform für neue SIP-basierte Mehrwertdienste. In Informationstechnische Gesellschaft im VDE (ITG) (Hrsg.), *Mobilfunk - Technologien und Anwendungen (ITG-Fachbericht Band 202)*. VDE Verlag GmbH, Mai 2007, S. 109–113.
- [Cris09] Chad Criswell. *What Are Ringback Tones?* suite101.com, http://cell-phones.suite101.com/article.cfm/what_are_ringback_tones, Februar 2009. (Abgerufen am 20.09.2009).

- [CSSW07] Rebecca Chen, Elisa CY Su, Victor Shen und Yi-Hong Wang. *Introduction to IP Multimedia Subsystem (IMS), Part 2: Building an IMS infrastructure*. IBM, <http://www.ibm.com/developerworks/webservices/library/ws-soa-ipmultisub2/>, Juni 2007. (Abgerufen am 10.11.2009).
- [Detk07] Evren Eren; Kai-Oliver Detken. *VoIP Security. Konzepte und Lösungen für sichere VoIP-Kommunikation*. Hanser Fachbuch. 1. Auflage, August 2007.
- [digi09] digium, <http://www.digium.com/en/products/software/skypeforasterisk.php>. *Skype For Asterisk*, Oktober 2009. (Abgerufen am 17.11.2009).
- [Döll09] Mirko Dölle. *Die Woche: Android als Standard-Smartphone-Linux?* heise online, <http://www.heise.de/open/artikel/Die-Woche-Android-als-Standard-Smartphone-Linux-851721.html>, November 2009. (Abgerufen am 07.11.2009).
- [Drlj09] Srdan Dido Drljaca. *IMS Subsystem - NGN*. <http://knol.google.com/k/sran-io-drljaa/ims-subsystem-ngn/3m85cpdhj12vd/16>, April 2009. What is IMS? How it works? NGN is not just VoIP and video, it is much much more. (Abgerufen 29.08.2009).
- [Dude06] Dudenredaktion (Hrsg.). *Duden. Deutsches Universalwörterbuch. Mit CD-ROM: Das umfassende Bedeutungswörterbuch der deutschen Gegenwartssprache*. Bibliographisches Institut, Mannheim. 6. überarb. u. erw. Auflage, Oktober 2006.
- [Duvu08] Sreeram Duvur. *Introducing GlassFish Communications Server*. http://blogs.sun.com/sdudv/entry/introducing_glassfish_communications_server, Dezember 2008. (Abgerufen am 24.11.2009).
- [dVel07] Thierry Van de Velde. *Value-Added Services for Next Generation Networks (Informa Telecoms & Media)*. Auerbach Publications. 1. Auflage, Dezember 2007.

- [Dysz07] Bill Dyszel. *Outlook 2007 für Dummies (Für Dummies)*. Wiley-VCH. 1. Auflage, Juni 2007.
- [Eric06] Ericsson Media Relations, <http://www.ericsson.com/ericsson/press/releases/20061205-1091558.shtml>. *Ericsson brings world's first live IMS network over 3G to Softbank Mobile in Japan (Press Release)*, Dezember 2006. (Abgerufen am 09.10.2009).
- [Eric08] Ericsson, <http://www.ericsson.com/solutions/page.asp?ArticleId=C31A70E1-A7CD-4817-9E16-8257F5765034>. *Ericsson IMS (IP Multimedia Subsystem)*, 2008. (Abgerufen am 15.11.2009).
- [Frau07] Fraunhofer FOKUS, http://www.fokus.fraunhofer.de/en/fokus/_pdfs/n113-db-ims-11-2007.pdf. *Beyond the Open Source IMS - Add-on components around the FOKUS Open Source IMS Core*, November 2007.
- [Frau09] Fraunhofer FOKUS, <http://www.fokus.fraunhofer.de/en/ngni/downloads/flyer/index.html>. *Open SOA Telco Playground Integrated Service Architectures (FOKUSinnovation Report)*, 10-2009. Auflage, Oktober 2009.
- [Gies09] Hartmut Gieselmann. *Xbox 360 streamt Full-HD-Videos ohne Wartezeit*. heise online, <http://www.heise.de/newsticker/meldung/Xbox-360-streamt-Full-HD-Videos-ohne-Wartezeit-861637.html>, November 2009. (Abgerufen am 17.11.2009).
- [Goog09] Google Inc., <http://www.google.com/mobile/navigation/>. *Google Maps Navigation (Beta)*, November 2009. (Abgerufen am 22.11.2009).
- [GüRu08] Sebastian Güttner und Daniel Ruf. *Aufbau eines VoIP-Netzes im Fachgebiet Kommunikationsnetze*. Medienprojekt, TU Ilmenau, FG Kommunikationsnetze, Mai 2008.
- [Haig05] Sven Haiges. *Einführung in die JAIN/SLEE-Technologie - Aufbruch in neue Telekommunikations-Welten*. entwickler.de, <http://it-republik.de/zonen/portale/psecom,id,101,online,908,neu,1.html>, Mai 2005. (Abgerufen am 20.11.2009).

- [Hill07] Juergen Hill. *Cisco TelePresence im Test: Das Ende für Bonus-Meilen?* Computerwoche, <http://www.computerwoche.de/593035>, Mai 2007. (Abgerufen am 21.11.2009).
- [IBM09] IBM. *WebSphere IP Multimedia Subsystem Connector*. IBM Deutschland, http://www-142.ibm.com/software/dre/ecatalog/detail.wss?locale=de_DE&synkey=M209950J71084S08, August 2009. IMS-Addon für WebSphere Application Server (Abgerufen am 22.08.2009).
- [ITU06] International Telecommunication Union, <http://www.itu.int/ITU-T/ngn/fgngn/>. *Focus Group on Next Generation Networks (FGNGN)*, Februar 2006.
- [ITU09] International Telecommunication Union, <http://www.itu.int/ITU-T/ngn/index.html>. *Next Generation Networks Global Standards Initiative (NGN-GSI)*, November 2009. (Abgerufen am 07.11.2009).
- [jNet09] jNetX. *jNetX' Vision of a Next Generation Service Delivery Model*. <http://jnetx.com/index.php?id=370>, Juli 2009. (Abgerufen am 25.08.2009).
- [Jo+ot09] Olle E. Johansson und andere. *Asterisk manager API*. <http://www.voip-info.org/wiki/view/Asterisk+manager+API>, Juli 2009. Voip-Info.org Wiki (Abgerufen am 22.08.2009).
- [JoTe07] William Jones und Jaime Teevan. *Personal Information Management*. University of Washington Press. Oktober 2007.
- [KaKö91] Andreas Kanbach und Andreas Körber. *ISDN. Die Technik*. Hüthig. 2. überarb. Aufl.. Auflage, 1991.
- [Klaß09] Christian Klaß. *Skype über SIP-Telefonanlagen nutzen - Skype-für-SIP-Betaprogramm gestartet*. Golem.de, <http://www.golem.de/0912/71624.html>, Dezember 2009. (Abgerufen am 03.12.2009).
- [KoMS07] Sascha Koesch, Fee Magdanz und Robert Stadler. *Massenvernetzung - Die Welt drängt ans Handy*. SPIEGEL ONLINE, <http://www.spiegel.de/netzwelt/mobil/0,1518,504283,00.html>, September 2007. (Abgerufen am 03.09.2009).

- [KuHi06] Akira Kurokawa und Isao Higashi. *Standardization Trends of the Next Generation Network in ETSI TISPAN*. NTT Technical Review, 2006.
- [LeTO07a] Armin Lehmann, Ulrich Trick und Steffen Oehler. NGN und Mehrwertdienste - Herausforderung und Chance. *ntz - VDE Verlag* Band 6, 2007, S. 22–25.
- [LeTO07b] Armin Lehmann, Ulrich Trick und Steffen Oehler. NGN und Mehrwertdienste - Technische Lösungen. *ntz - VDE Verlag* Band 7/8, 2007, S. 30–33.
- [LeTO08] Armin Lehmann, Ulrich Trick und Steffen Oehler. NGN und Mehrwertdienste - Geschäftsmodelle und Szenarien. *ntz - VDE Verlag* Band 1, 2008, S. 22–25.
- [Lewa07] Agnieszka Lewandowska. *Architektur für die Verarbeitung von Kontextinformationen: Architekturkonzept*. TU Ilmenau, FG Kommunikationsnetze, September 2007. Interner Forschungsbericht.
- [LuWi07] Stefan Lutherdt und Harmut Witte. *TAS - Touristisches Assistenzsystem für Urlaub, Freizeit- und Bildungsaktivitäten : Ein InnoRegio-Projekt der TU Ilmenau*. TU Ilmenau, FG Biomechatronik, 2007.
- [Mari08] Achim Marikar. Konzeption und Entwicklung einer multimedialen Dienstplattform für Next Generation Networks. Master Thesis, Fachhochschule Köln, Institut für Nachrichtentechnik, Januar 2008.
- [Mart08] Eduardo Martins. *Quick Start Guide with Maven 2 and Mobicents Sources*. mobicents-public, <http://groups.google.com/group/mobicents-public/web/quick-start-guide-with-maven-2>, Februar 2008. (Abgerufen am 28.10.2008).
- [MHSp09] MHSpot LLC, <http://www.mhspot.com/sts/siptosis.html>. *SipToSis - SIP to Skype Gateway Bridge Proxy Adapter Converter Interconnect*, April 2009. (Abgerufen am 29.06.2009).
- [MIKO⁺07] Naotaka Morita, Hideo Imanaka, Osamu Kamatani, Takumi Oba und Koji Tanida. *Overview and Status of NGN Standardization Activities at ITU-T*. NTT Technical Review, 2007.

- [Mine08] Anthony Minessale. *How does FreeSWITCH compare to Asterisk?* <http://www.freeswitch.org/node/117>, April 2008. (Abgerufen am 15.06.2009).
- [MSIR⁺09] Jared Morgan, Douglas Silas, Ivelin Ivanov, Vladimir Ralev, Eduardo Martins, Jean Deruelle, Oleg Kulikov, Amit Bhayani, Luis Barreiro, Alexandre Mendonça, Bartosz Baranowski, und Pavel Sleg. *JAIN SLEE Server User Guide*. Mobicents, <http://hudson.jboss.org/hudson/view/Mobicents/job/MobicentsBooks/lastSuccessfulBuild/artifact/jain-slee/index.html>, 3. Auflage, Juni 2009.
- [Open08] OpenCloud. *OpenCloud Rhino Real-time Application Server*. OpenCloud Ltd., <http://www.opencloud.com/products/rhino-application-server/real-time-application-server>, 2008. A JAIN SLEE based Application Server (Abgerufen am 25.08.2009).
- [Opit09] Rudolf Opitz. *Evolution der Netze - Die Zukunft des mobilen Surfens*. heise mobil, <http://www.heise.de/mobil/artikel/Die-Zukunft-des-mobilen-Surfens-222569.html>, Juni 2009. (Abgerufen am 17.08.2009).
- [Poll09] Klaus Pollak. *Wie Festnetzbetreiber mithilfe von Next-Generation-Networks aus der Krise herauskommen können*. network Computing, <http://www.networkcomputing.de/wie-festnetzbetreiber-mithilfe-von-next-generation-networks-aus-der-krise-herauskommen-koennen/>, März 2009. (Abgerufen am 20.09.2009).
- [Rale09] Vladimir Ralev. *Introducing Mobicents IPBX*. Vladimir Ralev's Blog, <http://vladimirralev.blogspot.com/2009/10/introducing-mobicents-ipbx.html>, Oktober 2009. (Abgerufen am 01.11.2009).
- [Robe09] Lawrence G. Roberts. A Radical New Router. *IEEE spectrum* Band 07.09, Juli 2009, S. 30–35.
- [Schm06] Jürgen Schmidt. Der Lochtrick - Wie Skype & Co. Firewalls umgehen. *c't* Band 17, 2006, S. 142.

- [Seit07] Jochen Seitz. *Vermittlungstechnik - Kommunikationsnetze*. TU Ilmenau, FG Kommunikationsnetze, April 2007. (Vorlesung).
- [Sieg01] Gerd Siegmund. *Intelligente Netze. Technik, Dienste, Vermarktung*. Hüthig Telekommunikation. 2., neubearb. u. erw. Auflage, 2001.
- [Sieg02] Gerd Siegmund. *Next Generation Networks. IP-basierte Telekommunikation*. Hüthig. 1. Auflage, 5 2002.
- [Siet09] Richard Sietmann. Der Stille Machtkampf - Next Generation Networks: Wie sich Netzbetreiber und Ausrüster die Zukunft der Telekommunikationsnetze vorstellen. *c't* Band 24, November 2009, S. 90–97.
- [Sila08] Douglas Silas. *Mobicents User Guide*. Red Hat Documentation Group, http://hudson.jboss.org/hudson/job/MobicentsDocumentation/lastSuccessfulBuild/artifact/html_single/index.html, Februar 2008. Mobicents is a highly scalable event-driven application server with a robust component model. (Abgerufen am 15.08.2009).
- [Stee07] David Steel. *Jukebox by text message*. Springwise, http://springwise.com/telecom_mobile/jukebox_by_text_message/, März 2007. (Abgerufen am 16.10.2009).
- [StMa08] Andreas Steger und Michael Mauser. Konzipierung und Aufbau eines FMC-fähigen Netzes. Medienprojekt, TU Ilmenau, FG Kommunikationsnetze, Mai 2008.
- [Sun04a] Sun Developer Network (SDP), <http://java.sun.com/products/jain/>. *JSLEE and the JAIN Initiative*, Dezember 2004. (Abgerufen am 14.11.2009).
- [Sun04b] Sun Microsystems. *JAIN and Java in Communications*, März 2004.
- [Tech09] Tech Invite, <http://www.tech-invite.com/Ti-etsi-standards-TISPAN.html>. *ETSI TISPAN NGN (Next Generation Network) Specifications*, September 2009. (Abgerufen am 08.11.2009).
- [Tour09] Benoit Touron. *Developping a Simple SIP application with Mobicents and Eclipse*. <http://ben-touron.blogspot.com/2009/11/>

- developping-simple-sip-application-with.html, November 2009. (Abgerufen am 30.11.2009).
- [Tric03] Ulrich Trick. All over IP - der Schlüssel zur ITK-Infrastruktur der Zukunft. *ntz - VDE Verlag* Band 1, 2003, S. 30–33.
- [TrWe07] Ulrich Trick und Frank Weber. *SIP, TCP/IP und Telekommunikationsnetze Next Generation Networks und VoIP-konkret, m. CD-ROM*. Oldenbourg. 3. überarb. u. erw. Auflage, 5 2007.
- [Voip09] Voip-info.org, <http://www.voip-info.org/wiki/view/Asterisk+SIP+Messaging>. *Asterisk SIP Messaging*, März 2009. (Abgerufen am 04.11.2009).
- [WeVi08] Peter Weik und Dragos Vingarzan. *An Open Source Reference Implementation for the Telecommunications Industry*. Fraunhofer FOKUS, 2008.
- [Wint09a] Stefan Wintermeyer. *Asterisk 1.4 + 1.6: Installation, Programmierung und Betrieb*. Addison-Wesley Longman Verlag. Onlineausgabe: [Wint09b], 1. Auflage, Juni 2009.
- [Wint09b] Stefan Wintermeyer. *Asterisk 1.4 + 1.6: Installation, Programmierung und Betrieb*. <http://www.das-asterisk-buch.de/2.1/>, Juni 2009. (Abgerufen am 13.09.2009).
- [Zieg09] Peter-Michael Ziegler. *Studie: Echtzeit-Unterhaltung ist Web-Traffic-Größe Nummer 1*. heise online, <http://www.heise.de/newsticker/meldung/Studie-Echtzeit-Unterhaltung-ist-Web-Traffic-Groesse-Nummer-1-839567.html>, Oktober 2009. (Abgerufen am 1.11.2009).
- [Ziva09] Dusan Zivadinovic. Betreutes Funken - Mini-Basisstationen verbessern Handy-Telefonate. *c't* Band 18, August 2009, S. 146 – 147.
- [Zmor09] Amir Zmora. *Bringing Telephony Features into SIP Networks with Back To Back User Agent*. RADVISION (Product Marketing, SIP Products), <http://www.sipcenter.com/sip.nsf/html/Bringing+Telephony+Features+into+SIP+Networks+with+Back+To+Back+User+Agent>, August 2009. (Abgerufen am 07.11.2009).

Standards und Empfehlungen

- [3GPP23228] 3rd Generation Partnership Project. *Technical Specification Group Services and System Aspects; IP Multimedia Subsystem (IMS) Stage 2, Release 9 (3GPP Technical Specification TS 23.228)*, Juni 2009.
- [3GPP23979] 3rd Generation Partnership Project. *Technical Specification Group Services and System Aspects; 3GPP enablers for Open Mobile Alliance (OMA); Push-to-talk over Cellular (PoC) services (3GPP Technical Report TR 23.979)*, Dezember 2008.
- [AC3] Advanced Television Systems Committee, Inc., http://www.atsc.org/standards/a_52b.pdf. *Digital Audio Compression Standard (AC-3, E-AC-3)*, Juni 2005.
- [ETSI181001] European Telecommunications Standards Institute. *Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN) NGN Release 1 - Release definition (ETSI Technical Report TR 181 001)*, März 2006.
- [ETSI181003] European Telecommunications Standards Institute. *Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN) Services Capabilities, Requirements and strategic direction for NGN services (ETSI Technical Report TR 181 003)*, Mai 2006.
- [ETSI181004] European Telecommunications Standards Institute. *Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN) NGN Generic capabilities and their use to develop services (ETSI Technical Report TR 181 004)*, März 2006.
- [G114] International Telecommunication Union. *International telephone connections and circuits - General Recommendations on the transmission quality for an entire international telephone connection - One-way transmission time (ITU-T Recommendation G.114)*, Mai 2003.

- [H323] International Telecommunication Union, <http://www.itu.int/rec/T-REC-H.323-200606-I/en>. *Infrastructure of audiovisual services - Systems and terminal equipment for audiovisual services - Packet-based multimedia communications systems (ITU-T Recommendation H.323)*, Juni 2006.
- [JSR22] Java Community Progress Program (JCP), <http://www.jcp.org/en/jsr/detail?id=22>. *Java Specification Requests - JSR 22: JAIN Service Logic Execution Environment API Specification*, März 2004. (Abgerufen am 14.11.2009).
- [JSR240] Java Community Progress Program (JCP), <http://www.jcp.org/en/jsr/detail?id=240>. *Java Specification Requests - JSR 240: JAIN SLEE (JSLEE) v1.1*, Juli 2008. (Abgerufen am 14.11.2009).
- [JSR281] Java Community Progress Program (JCP), <http://jcp.org/en/jsr/detail?id=281>. *Java Specification Requests - JSR 281:IMS Services API*, April 2009. (Abgerufen am 15.11.2009).
- [JSR289] Java Community Progress Program (JCP), <http://jcp.org/en/jsr/detail?id=289>. *Java Specification Requests - JSR 289: SIP Servlet v1.1*, August 2008. (Abgerufen am 14.11.2009).
- [JSR315] Java Community Progress Program (JCP), <http://www.jcp.org/en/jsr/detail?id=315>. *Java Specification Requests - JSR 315: Java Servlet 3.0 Specification*, Mai 2009. (Abgerufen am 14.11.2009).
- [JSR325] Java Community Progress Program (JCP), <http://jcp.org/en/jsr/detail?id=325>. *Java Specification Requests - JSR 325: IMS Communication Enablers (ICE)*, September 2009. (Abgerufen am 15.11.2009).
- [RFC2251] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc2251>. *Lightweight Directory Access Protocol (v3) (RFC 2251 IETF Network Working Group)*, Dezember 1997.
- [RFC2543] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc2543>. *SIP: Session Initiation Protocol (RFC 2543 IETF Network Working Group)*, März 1999. (veraltet, ersetzt durch [RFC3261, RFC3262, RFC3264, RFC3265]).

- [RFC2960] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc2960>. *Stream Control Transmission Protocol (RFC 2960 IETF Network Working Group)*, Oktober 2000. (veraltet, ersetzt durch RFC 4960).
- [RFC3050] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc3050>. *Common Gateway Interface for SIP (RFC 3050 IETF Network Working Group)*, Januar 2001.
- [RFC3261] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc3261>. *SIP: Session Initiation Protocol (RFC 3261 IETF Network Working Group)*, Juni 2002.
- [RFC3262] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc3262>. *Reliability of Provisional Responses in the Session Initiation Protocol (SIP) (RFC 3262 IETF Network Working Group)*, Juni 2002.
- [RFC3264] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc3264>. *An Offer/Answer Model with the Session Description Protocol (SDP) (RFC 3264 IETF Network Working Group)*, Juni 2002.
- [RFC3265] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc3265>. *Session Initiation Protocol (SIP)-Specific Event Notification (RFC 3265 IETF Network Working Group)*, Juni 2002.
- [RFC3344] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc3344>. *IP Mobility Support for IPv4 (RFC 3344 IETF Network Working Group)*, August 2002.
- [RFC3435] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc3435>. *Media Gateway Control Protocol (MGCP) Version 1.0 (RFC 3435 IETF Network Working Group)*, Juni 2002.
- [RFC3863] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc3863>. *Presence Information Data Format (PIDF) (RFC 3863 IETF Network Working Group)*, August 2004.
- [RFC3880] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc3880>. *Call Processing Language (CPL) - A Language for User Control of Internet Telephony Services (RFC 3880 IETF Network Working Group)*, Oktober 2004.

- [RFC3920] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc3920>. *Extensible Messaging and Presence Protocol (XMPP): Core (RFC 3863 IETF Network Working Group)*, Oktober 2004.
- [RFC3986] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc3986>. *Uniform Resource Identifier (URI): Generic Syntax (RFC 3986 IETF Network Working Group)*, Januar 2005.
- [RFC4353] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc4353>. *A Framework for Conferencing with the Session Initiation Protocol (SIP) (RFC 4353 IETF Network Working Group)*, Februar 2006.
- [RFC4566] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc4566>. *SDP: Session Description Protocol (RFC 4566 IETF Network Working Group)*, Juli 2006.
- [RFC5545] Internet Engineering Task Force, <http://tools.ietf.org/html/rfc5545>. *Internet Calendaring and Scheduling Core Object Specification (iCalendar) (RFC 5545 IETF Network Working Group)*, September 2009.
- [Y2001] International Telecommunication Union. *Global Information Infrastructure, Internet Protocol Aspects and Next-Generation Networks - General overview of NGN (ITU-T Recommendation Y.2001)*, Dezember 2004.
- [Y2011] International Telecommunication Union. *Global Information Infrastructure, Internet Protocol Aspects and Next-Generation Networks - Next Generation Networks - Frameworks and functional architecture models (ITU-T Recommendation Y.2011)*, Oktober 2006.
- [Y2012] International Telecommunication Union. *Global Information Infrastructure, Internet Protocol Aspects and Next-Generation Networks - Next Generation Networks - Instantiation of NGN reference points (ITU-T Recommendation Y.2012)*, August 2006.

Abbildungsverzeichnis

1.1	Entwicklung der Sprach- und Datenkommunikation	1
2.1	Struktur eines Telekommunikationsnetzes	3
2.2	ISDN-Fernnetz	6
2.3	Aufbau GSM-Netz	7
2.4	Datenverkehr in Mobilfunknetzen	9
2.5	Intelligente Netzknoten	10
2.6	IN-Dienstklassen	11
2.7	Breitbandstandards für Zugangsnetz	13
2.8	NGN vs. herkömmliche Netze	14
3.1	NGN-Zeitleiste	16
3.2	NGN-Modell der ITU-T	18
3.3	Aufbau IMS (3GPP)	20
3.4	ETSI Arbeitsgruppen und NGN-Spezifikationen	21
3.5	Aufbau IMS (ETSI)	22
3.6	OSI-Schichtenmodell	24
3.7	Protokoll-Stack SIP	26
3.8	Three-Way-Handshake SIP-Verbindungsaufbau	30
3.9	Aufbau SIP-Verbindung über Proxy	31
3.10	Protokollverschachtelung RTP	35
3.11	Softphones (UA)	36
3.12	Stateless und Statefull Proxy	37
3.13	Back-to-back UA	39
3.14	Konferenzserver	40
3.15	Zugangsnetze im NGN	43
3.16	Bandbreite in kabelgebundenen und kabellosen Breitbandnetzen	48
4.1	Dienste in den OSI-Schichten	50
4.2	Klassifikation von Diensten	51

4.3	Providertypen im NGN	52
4.4	Betriebsmodi eines Anwendungsservers	54
4.5	Darstellung vs. Inhalte	56
4.6	Device Unify Service	58
4.7	Device-Unify-Service technische Beschreibung	58
4.8	Push-to-talk over Cellular	60
4.9	T-Home Set-Top-Box	65
4.10	IPTV Call Notifier	67
4.11	Personal Contact Pages	69
4.12	Multichannel-Sitzungen	70
4.13	Combinatory Services	72
4.14	Videokonferenzsystem	73
4.15	Alarmierungsdienst	74
4.16	Abfragedienst	78
4.17	Mediencontroller	79
4.18	Meet-me-Service	82
4.19	Übersicht über die untersuchten Dienstklassen und Dienstkonzepte	84
5.1	Service Delivery Platform	85
5.2	Anwendungsserver als unabhängiges Netzelement	87
5.3	Betriebsmodus Content-Delivery	88
5.4	Betriebsmodus Wake-Up	89
5.5	Betriebsmodus Call Forwarding	90
5.6	Betriebsmodus Click-to-dial	90
5.7	APIs für die Dienststeuerung im NGN	91
5.8	Low- vs. High-Level-API	92
5.9	Architektur der Parlay 1.2 API	93
5.10	JAIN-Plattform	94
5.11	Ablauf Call-Processing-Language-Anfrage	96
5.12	Komponenten des Open Source IMS Playgrounds	97
5.13	Websphere IMS Connector	99
5.14	Mobicents Plattform	99
5.15	OpenCoud Rhino Application Server	100
5.16	SailFin Anwendungsserver	101
6.1	Aufbau bisheriges FMC-Netz	105
6.2	Softphones (UA)	107

6.3	Sinus 501v (UA)	108
6.4	Aufbau von Asterisk	111
6.5	SipToSis Skype-Gateway	113
6.6	SipToSis - Eingehendes Gespräch	114
6.7	Netzkonzept	115
7.1	Diaplan Asterisk	118
7.2	Mobicents High-Level-Diagramm	121
7.3	JAIN SLEE Service-Building-Blocks	122
7.4	SIP Servlet Application Router	123
7.5	Management-Konsolen Mobicents	124
7.6	Aktivitätsdiagramm Ring-Back-When-Free	125
7.7	Realisierung Combinatory Services	127
7.8	Realisierung Click-to-Dial	128
7.9	SIP TAPI Setup	129
7.10	Realisierung Mediencontroller	130
7.11	IPTV-Call-Notifier	131
7.12	Überblick Funktionsblöcke	137
A.1	Eclipse IDE für Java	152
A.2	Repository Browser Eclipse	153
A.3	Mobicents Plugins für Eclipse	155
B.1	Klassifizierungen für Dienste im NGN (detailliert)	158

Tabellenverzeichnis

3.1	Sprachcodecs und Kennwerte	34
5.1	Vergleich Softwareplattformen Anwendungsserver	104

Abkürzungsverzeichnis und Formelzeichen

3GPP	3 rd G eneration P artnership P roject
AAA	A uthentication, A uthorization, A ccounting-System
AEL	A sterisk E xtension L anguage
AGI	A sterisk G ateway I nterface
AMI	A sterisk M anagement I nterface
API	A pplication P rogramming I nterface
AS	A nwendungsserver
ATM	A synchronous T ransfer M ode
B2BUA	B ack- 2 - B ack U ser A gent
CAMEL	C ustomized A pplication for M obile Network E nhancement L ogic
CAP	C amel A pplication P art
CGI	C ommon G ateway I nterface
CPL	C all P rocessing L anguage
CSCF	C all and S ession C ontrol F unctions
CSE	C amel S ervice E nvironment
DFN	D eutsches F orschungsnetzwerk
DOCSIS	D ata O ver C able S ervice I nterface S pecification
DSL	D igital S ubscriber L ine
DSLAM	D SL A ccess M ultiplier
DSS1	D igital S ubscriber S ignalling S ystem No. 1
DTMF	D ual-tone m ulti- f requency
EDGE	E nhanced D ata Rates for G SM E volution
ETSI	E uropean T elecommunications S tandards I nstitute
FMC	F ixed M obile C onvergence
FTP	F ile T ransfer P rotocol
GPL	G NU G eneral P ublic L icense
GPRS	G eneral P acket R adio S ervice

GPS	G lobal P ositioning S ystem
GSM	G lobal S ystem for M obile C ommunications
HSDPA	H igh S peed D ownlink P acket A ccess
HSS	H ome S ubscriber S ervice
HSUPA	H igh S peed U plink P acket A ccess
HTTP	H ypertext T ransfer P rotocol
IETF	I nternet E ngineering T ask F orce
IM	I ntant M essaging
IMS	I P M ultimedia S ubsystem
IMSI	I nternational M obile S ubscriber I dentify
IN	I ntelligentes N etz
INAP	I ntelligent N etwork A pplication P art
IP	I nternet P rotocol
IPTV	I nternet P rotocol T elevision
ISDN	I ntegrated S ervices D igital N etwork
ITU	I nternational T elecommunication U nion
JAIN	J ava A PIs for I ntegrated S ervice
JavaEE	J ava E nterprise E dition
JDK	J ava D evelopment K it
JRE	J ava R untime E nvironment
JSLEE	J AIN S ervice L ogic E xecution E nvironment
LBS	L ocation B ased S ervices
LDAP	L ightweight D irectory A ccess P rotocol
LTE	L ong T erm E volution
MGW	M edia G ateway
MMS	M ultimedia M essaging S ervice
MPLS	M ultiprotocol L abel S witching
NGN	N ext G eneration N etwork
OMA	O pen M obile A lliance
OMAP	O perations, M aintenance and A dmistration P art
OSA	O pen S ervice A ccess
OSE	O MA S ervice E nvironment
PBX	P ublic B ranch E xchange
PIM	P ersonal I nformation M anager
PoC	P ush-to-talk o ver C ellular
POTS	P lain O ld T elephone S ervice

PSTN	P ublic S witched T elephone N etwork
RA	R essource A dapter
RSS	R eally S imple S yndication
RSVP	R esource R eservation P rotocol
RTCP	R eal- T ime C ontrol P rotocol
RTP	R eal- T ime T ransport P rotocol
SCTP	S tream C ontrol T ransmission P rotocol
SDP	S ervice D elivery P latform
SDP	S ession D escription P rotocol
SGW	S ignaling G ateway
SIM	S ubscriber I dentify M odule
SIP	S ession I nitiation P rotocol
SLEE	S ervice L ogic E xecution E nvironment
SMS	S hort M essage S ervice
SOA	S ervice O riented A rchitecture
SPIT	S pam via I nternet T elephony
SS7	S ignaling S ystem N o. 7
TCP	T ransmission C ontrol P rotocol
TISPAN	T elecoms and I nternet converged S ervices and P rotocols for Advanced Networks
TMC	T raffic M essage C hannel
TTS	T ext- t o- S peech
TVst	T eil- V ermittlungs s telle
UA	U ser A gent
UDP	U ser D atagram P rotocol
UML	U nified M odelling L anguage
UMTS	U niversal M obile C ommunications S ystem
URI	U niform R essource I dentifier
URL	U niform R essource L ocator
VAS	V alue A dded S ervices (Mehrwertdienste)
VoD	V ideo- o n- D emand
VPN	V irtual P rivate N etwork
Vst	V ermittlungs s telle
WiMAX	W orldwide I nteroperability for M icrowave A ccess
WLAN	W ireless L ocal A rea N etwork
XML	E xtensible M arkup L anguage

Thesen zur Diplomarbeit

1. Das Next Generation Network (NGN) ist ein IP-basierendes Kommunikationsnetz, welches verschiedene Zugangsnetze (WLAN, UMTS, xDSL etc.) zu einem Netz mit einheitlichen Steuer- und Transportfunktionen zusammenfasst.
2. Neben der Einbindung von anderen Kernnetzen (ISDN, GSM) bietet das NGN eine Reihe weiterer Leistungsmerkmale (Konferenzen, indirekte Sitzungssteuerung) und Medientypen (Audio, Video, Text etc.).
3. Die grundlegenden Eigenschaften des NGN werden anhand der Standardisierungsbemühungen von ITU-T, ETSI und 3GPP vorgestellt.
4. Die neue Netzstruktur und die Multimediaunterstützung ermöglichen eine Vielzahl innovativer Mehrwertdienste im NGN. Parallel zur Untersuchung dieser Dienste werden Konzepte zur Umsetzung erarbeitet.
5. Zur Erzeugung dieser Mehrwertdiensten existieren mehrere standardisierte Schnittstellen und Plattformen, die analysiert und bewertet werden.
6. Für etliche Schnittstellen existieren bereits freie und kommerzielle Implementierungen.
7. Nach der Konzeption und dem Aufbau eines lokalen NGN wird ein Anwendungsserver ausgewählt und eingerichtet.
8. Auf Basis des Anwendungsservers und des lokalen NGN werden verschiedene Dienstkonzepte prototypisch umgesetzt.

Ilmenau, den 07. 12. 2009

Torsten Strelle

Erklärung

Die vorliegende Arbeit habe ich selbstständig ohne Benutzung anderer als der angegebenen Quellen angefertigt. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten Quellen entnommen wurden, sind als solche kenntlich gemacht. Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer oder anderer Prüfungen noch nicht vorgelegt worden.

Ilmenau, den 07. 12. 2009

Torsten Strelle