(19) **Bundesrepublik Deutschland**
Deutsches Patent- und Markenamt

(10) **DE 694 32 287 T2** 2004.01.15

(12) # Übersetzung der europäischen Patentschrift

(97) **EP 0 660 576 B1**
(21) Deutsches Aktenzeichen: **694 32 287.3**
(96) Europäisches Aktenzeichen: **94 309 352.6**
(96) Europäischer Anmeldetag: **14.12.1994**
(97) Erstveröffentlichung durch das EPA: **28.06.1995**
(97) Veröffentlichungstag der Patenterteilung beim EPA: **19.03.2003**
(47) Veröffentlichungstag im Patentblatt: **15.01.2004**

(51) Int Cl.[7]: **H04M 11/06**
H04M 3/50, H04Q 11/04, G09G 5/00

(30) Unionspriorität:
171311 **21.12.1993** **US**

(84) Benannte Vertragsstaaten:
DE, GB

(73) Patentinhaber:
AT & T Corp., New York, N.Y., US

(74) Vertreter:
derzeit kein Vertreter bestellt

(72) Erfinder:
Garland, J. David, Berkeley Heights, New Jersey 07922, US; McGee, Andrew R., South Plainfield, New Jersey 07080, US

(54) Bezeichnung: **Multimediensystem**

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel II § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

Beschreibung

TECHNISCHES GEBIET

[0001] Die Erfindung betrifft ein Multimedia-Telekommunikationssystem.

HINTERGRUND DER ERFINDUNG

[0002] Als Folge der Implementierung von ISDN in dem öffentlichen Telekommunikationsnetz ist eine Anzahl von neuen Diensten eingeführt worden. Ein derartiger Dienst beinhaltet Multimedia-Anwendungen, die Übertragung von Video (d. h. Bildern) und entsprechenden Sprachsignalen. Als Folge von Multimedia-Anwendungen können sich eine anrufende und eine angerufene Partei (d. h. Gegenstelle) während einer Telefonkonversation über einem ISDN-Kommunikationspfad sehen und hören. Multimedia umfasst andere Dienstanwendungen, die an einem Anrufer über das ISDN angeboten werden können, z. B. Anwendungen, die einen Anrufer unterhalten, beispielsweise ein Bewegungsvideo mit begleitendem Ton. Es sei darauf hingewiesen, dass derartige Multimedia-Dienstanwendungen höchstwahrscheinlich von verschiedenen Multimediaanwendungsanbietern entwickelt werden: Wenn es sich herausstellt, dass dies der Fall ist, dann würde es wünschenswert sein, einem Benutzer zu erlauben, auf unterschiedliche Multimediaanwendungen zuzugreifen, die von unterschiedlichen Anbietern davon entwickelt werden, und zwar unabhängig von dem Typ von Gerät, z. B. Computerterminal, das der Benutzer zum Durchführen eines derartigen Zugriffs verwendet.

[0003] E. J. Addeo et al. offenbaren in „An experimental multi-media bridging system", Sigois Bulletin, Vol. 99 No. 2–3, auf den Seiten 236 242 (1988) ein Multimedia-Überbrückungssystem, das eine Mehrfachablage-(multi shelf)-Architektur umfasst. Jede Ablage entspricht einem bestimmten Medium, so dass das System z. B. eine „Video-Ablage", eine „Audio-Ablage", eine „Graphik-Ablage" etc. umfasst. Jede Ablage (shelf) verarbeitet und schaltet das Signal entsprechend zu seinem jeweiligen Medium unter der Steuerung einer Gesamtsteuerschicht.

[0004] Multimediaszenen werden derart aufgeteilt, dass die „Video-Ablage" des Überbrückungssystems Videodaten enthalten wird, die sich auf sämtliche Multimediaszenen beziehen, die „Audio-Ablage" die Audiodaten enthalten wird, die sich auf sämtliche Multimediaszenen beziehen, die „Graphik-Ablage" die Graphikdaten enthalten wird, die sich auf sämtliche Szenen etc. beziehen.

[0005] Es sei darauf hingewiesen, dass eine Entwicklung einer Multimediaanwendung nicht eine einfache Aufgabe ist und typischerweise die Konstruktion der, Anwendung, ein „debugging" davon und dann das Speichern davon auf einem Multimediasystem zur Darbietung an einen Benutzer beinhaltet. Es ist oft der Fall, dass die Anwendung ein zusätzliches „debugging" benötigen wird, nachdem sie in dem Multimediasystem gespeichert worden ist, um Schnittstellenprobleme zu behandeln, die zwischen der neu konstruierten Multimediaanwendung und dem Multimediasystem auftreten können. Ein Konstrukteur/ Anbieter einer Multimediaanwendung kann das letztere Problem dadurch behandeln, dass das Multimediasystem dupliziert wird und das System als ein Werkzeug während der Konstruktions-/ Entwicklungs- und Debugging-Phase einer Multimediaanwendung verwendet wird. Es sei jedoch darauf hingewiesen, dass sich eine Duplizierung eines Multimediasystem für den Zweck dieses zum Konstruieren und Debuggen einer Multimediaanwendung zu verwenden, als kostenintensiv und etwas unnötig herausstellen könnte.

[0006] Die vorliegende Erfindung stellt ein Multimediasystem bereit, wie im Anspruch 1 beansprucht.

[0007] Bevorzugte Ausführungsformen der Erfindung werden nun mit Hilfe eines illustrativen Beispiels unter Bezugnahme auf die beiliegenden Zeichnungen beschrieben. In den Zeichnungen zeigen:

[0008] **Fig.** 1 ein Multimediasystem, in dem die Prinzipien der Erfindung umgesetzt werden können;

[0009] **Fig.** 2 bis 7 in einer Flussdiagrammform das Programm, welches die Prinzipien der Erfindung in dem Benutzerterminal der **Fig.** 1 implementiert;

[0010] **Fig.** 8 bis 21 in einer Flussdiagrammform das Programm, das die Prinzipien der Erfindung in dem Audioserver der **Fig.** 1 implementiert;

[0011] **Fig.** 22 bis 31 in einer Flussdiagrammform das Programm, das die Prinzipien der Erfindung in dem Datenserver der **Fig.** 1 implementiert.

[0012] **Fig.** 32 in einer Flussdiagrammform das Programm, das. die Prinzipien der Erfindung in dem Anwendungscontroller der **Fig.** 1 implementiert.

AUSFÜHRLICHE BESCHREIBUNG

[0013] Bezug nehmend nun auf **Fig.** 1 ist dort ein grobes Blockdiagramm eines Multimediasystem **100** gezeigt, das die Prinzipien der vorliegenden Erfindung verkörpert. Insbesondere umfasst das System **100** eine Anzahl von Controllern, die mit miteinander zusammen arbeiten, um eine bestimmte Multimediaanwendung einem Benutzer anzubieten. Derartige Controller umfassen den Datenserver **10**, den Anwendungscontroller **20**, den Audioserver **30**, die Arbeitsstation **40**, die Entwicklungsplattform **150** und das Betriebs-Unterstützungssystem **50**. Der Datenserver **10**, der z. B. die 386-WGS-Arbeitsstation (Work Station) sein kann, die von AT & T verfügbar ist, arbeitet mit dem MS DOS Betriebssystem und ist programmiert, um eine Datenfunktion im Zusammenhang mit der Darbietung einer Multimediaanwendung bereit zu stellen. Insbesondere ist der Datenserver **10** mit dem Benutzer über den Kommunikationspfad **11-1** gekoppelt und reagiert auf einen Empfang einer Aufforderung von dem Benutzer, indem er bestimmt,

ob er die Aufforderung selbst verarbeiten wird oder die Aufforderung an entweder den Anwendungscontroller **20** oder den Audioserver (Controller) **30** weiterleiten wird. Wenn z. B. die Aufforderung auf (a) Audio, beispielsweise eine bestimmte Auswahl von Musik, gerichtet ist, dann übergibt der Datenserver **10** die Aufforderung über das LAN **45** an den Audioserver **30** für eine Verarbeitung; oder auf (b) eine bestimmte Multimediaanwendung gerichtet ist, dann übergibt der Datenserver **10** die Aufforderung an den Anwendungscontroller **20** über das LAN **45**. Wenn sich die Aufforderung auf eine Datenfunktion bezieht, z. B. auf bestimmte Daten, die in dem Speicher **61** gespeichert sind, dann entlädt der Datenserver **10** die Daten von dem Speicher **61** und überträgt die Daten an den Benutzer über den Pfad **11-1**. In einer illustrativen Ausführungsform der Erfindung können derarige Daten z. B. digitalisierte Bilder, Text, Graphik-Primitive, Animations-Primitive, ausführbare Programme etc. sein, die in der Datenbank **61** gespeichert sind. In einer illustrativen Ausführungsform der Erfindung können derartige Daten in der Datenbank **61** in einem komprimierten Format gespeichert sein (wie nachstehend erläutert wird, wird die Dekompression von derartigen Daten von dem Terminal **T1** ausgeführt, bevor sie einem Benutzer dargeboten werden).

[0014] Die Schnittstelle, die der Datenserver **10** an dem Benutzer darbietet, arbeitet auch dazu, um den Empfang eines ankommenden Anrufs über den Pfad **11-1** zu erkennen. In einer illustrativen Ausführungsform der Erfindung kann der Pfad **11-1** zum Beispiel auch ein ISDN Primärraten-Schnittelle (Primary Rate Interface) Kommunikationspfad sein, der 23 B Kanäle und einen D Kanal umfasst. Wie altbekannt ist wird jeder B Kanal zum Transportieren eines Telefonanrufs (Sprache oder Daten) verwendet , und der D Kanal wird zum Weiterleiten von Signalisierungsinformation im Zusammenhang mit Anrufen, die über die B Kanäle transportiert werden, verwendet. Kommunikationspfade **11-2**, **11-3** und **152** können ebenfalls eine ISDN PRI sein.

[0015] Insbesondere dann, wenn der Benutzer einen Anruf an ein Multimediasystem **100** richtet, verarbeitet das Telefonnetz **200** den Anruf in einer herkömmlichen Weise und alarmiert das System **100** über den ankommenden Anruf über einen ISDN D Signalisierungskanal, der zu dem Datenserver **10** über den Pfad **11-1** reicht. Eine derartige Alarmierung umfasst die Benachrichtigung des Datenservers **10** über die Identität (Nummer) des ISDN B-Kanals, der den Anruf tragen wird. Der Server **10** wiederum reagiert auf eine derartige Signalisierung durch Senden einer Anrufakzeptierungsnachricht (call accept message) an das Netz **200** über den Signalisierungskanal, wodurch eine Kommunikationsverbindung zwischen dem System **100** und dem Benutzer hergestellt wird. An diesem Punkt können das System und der Benutzer **100** beginnen, Nachrichten miteinander auszutauschen, wie nachstehend erläutert wird.

[0016] Der Anwendungscontroller **20** ist insbesondere zum Ausführen von sämtlichen Multimediaanwendungen, die das System **100** (der Controller **20**) an das Terminal des Benutzers, z. B. das Terminal **T1**, sendet, verantwortlich. Das heißt, der Controller **20**, der zum Beispiel der 386-WGS Computer sein kann, der von AT & T verfügbar ist, arbeitet unter der Steuerung des Unix Betriebssystems, um eine Multimediaanwendung an dem Benutzer bereit zu stellen, wobei die Anwendung Daten und/oder Sprache (Audio) umfassen kann. Wie voranstehend erwähnt, können derartige Daten digitalisierte Bilder, Text, Graphik-Primitive, Animations-Primitive, ausführbare Programme etc., die in der Datenbank **61** gespeichert sind, umfassen. Der begleitende analoge Abschnitt von derartigen Daten wird in digitaler Form in der Datenbank **60** gespeichert. Für den Datenabschnitt einer Anwendung weist der Controller **20** den Datenserver **10** über das LAN **45** an, die Daten von dem Speicher **61** zu entladen und die Daten an den Benutzer über den Pfad **11-1** darzubieten. Für den Audioabschnitt der Anwendung weist der Controller **20** den Audioserver **30** über das LAN **45** an, die zugehörigen digitalen Audiodaten von dem Speicher **60** zu entladen und die Audiodaten an den Benutzer über einen ISDN Kanal des Pfads **11-2** darzubieten. Der Anwendungscontroller **20** kann den Server **10** (30) auch anweisen, einen Empfang einer Daten- (Audio) – Datei zu erwarten, die von dem Benutzer eingegeben werden soll, und die Datei im Speicher **61** (60) zu speichern. Der Anwendungscontroller **20** überträgt Information an den Benutzer, indem er diese Information auf dem LAN **45** für eine Lieferung an den Benutzer über den Datenserver **10** platziert. Eine derartige Information kann sich auf den Status einer angeforderten Anwendung oder eines Fehlers, der dazu gehört, beziehen.

[0017] Der Audioserver **30**, der ebenfalls ein 386-WGS sein kann und der ebenfalls unter dem Unix Betriebssystem arbeitet, stellt eine Schnittstelle zwischen dem Benutzer oder dem Anwendungscontroller **20** und dem Speicher **60** bereit. In der illustrativen Ausführungsform der Erfindung werden digitalisierte Audiodaten, die zu jeweiligen Anwendungen gehören, und/oder digitalisierte Audiodaten, die von einem Benutzer bereit gestellt werden, in dem Speicher **60** gespeichert. Der Audioserver **30** ist deshalb für die Speicherung und die „Abspielung" (das „Play Back") von Audiodaten, die zu einer jeweiligen Multimediaanwendung gehören, verantwortlich.

[0018] Das Betriebsunterstützungssystem (Operation's Support System, OSS) **50**, welches zum Beispiel des CompuLert System sein kann, das von AT & T erhältlich ist, überwacht den gesamten Betriebsstatus des Systems **100**. Insbesondere sind der Datenserver **10**, der Anwendungscontroller **20** und der Audioserver **30** angeordnet, um eine Statusnachricht an das OSS **50** über Kommunikationsstrecken **10-1**, **20-1** bzw. **30-1** periodisch zu senden. Derartige Statusnachrichten (die manchmal auch als „Herzschlag"-Nachrichten bezeichnet werden) zeigen dem

OSS an, dass der jeweilige Controller oder Server noch betriebsfähig ist. Wenn das OSS **50** keine derartige Nachricht innerhalb einer vorgegebenen Zeitperiode, z. B. einmal pro Minute, empfängt, dann nimmt das OSS **50** an, dass der Controller oder Server, die die Statusnachricht nicht gesendet haben, fehlerhaft sein können und benachrichtigt deshalb einen Administrator des Systems **100** durch Ausgeben einer geeigneten Nachricht an ein Systemterminal (nicht gezeigt). Wenn ein Controller oder Server einen Fehler selbst entdeckt, z. B. einen möglichen Fehler in dem Pfad **11-1**, dem Speicher **60** oder **61**, oder dem LAN **45**, darin berichtet der Controller oder Server dem OSS **50** diese Tatsache über eine jeweilige der Strecken **10-1**, **20-1** oder **30-1**. Das OSS **50** gibt wiederum eine Nachricht aus, die den Fehler an dem voranstehend erwähnten Systemterminal identifiziert.

[0019] Der Benutzer kann auf das System **100** durch Betreiben des Terminals **T1** zugreifen, das z. B. ein 386 oder 486 gestützter Prozessor sein kann, der unter der Steuerung des MS DOS Betriebssystems arbeitet. Insbesondere umfasst das Terminal **T1** eine Bibliothek von Multimediafunktionen, die zu dem System **100** gehören und eine herkömmliche Telefonieschaltuagsanordnung, so dass es mit dem Netz **200** gekoppelt werden kann. Eine derartige Schaltungsanordnung, die z. B. die Schaltungsplatine des Modells IDC sein kann, die von DGM & Sinc. aus Mount Lawel, New Jersey erhältlich ist, implementiert eine Schnittstelle für eine ISDN Basisratenschnittstelle (Basic Rate Interface, BRI), die in den Figuren durch den Kommunikationspfad **9** dargestellt wird. Derartige Funktionen umfassen unter anderem eine Dekomprimierung von Daten, die über einen beide oder BRI B-Kanä1e des Pfads **9** von dem System **100** empfangen werden. Die dekomprimierten Daten können dann auf der Anzeige **12** des Terminals **T1** angezeigt oder in. den Speicher innerhalb des **T1**, beispielsweise einem Floppy Disk Speicher oder einem Festplattenspeicher, gespeichert werden. In einer illustrativen Ausführungsform der Erfindung können die dekomprimierten Daten ein ausführbares Programm sein, das auf dem Terminal **T1** ablaufen gelassen werden soll, nachdem es gespeichert und dann in dem Terminal **T1** ausgeführt worden ist.

[0020] In **Fig.** 1 ist ersichtlich, dass das Terminal **T1** zu dem Stationssatz S1 und optional zu dem Audiosystem **5** gehört. Das Audiosystem **5** arbeitet insbesondere zum Darbieten von Audioinformation an dem Benutzer, in dem die Audioinformation von dem System **100** über irgendeinen der zwei B-Kanä1e des Pfads **9** empfangen wird. Alternativ kann ein Benutzer Audioinformation an dem System **100** für eine Speicherung in einem Speicher **60** darbieten, indem er in den Hörer der Station S1 spricht. Die Sprachsignale, die von der Station S1 empfangen werden, werden dann an der voranstehend erwähnten IDC Schaltungsanordnung dargeboten, wo sie digitalisiert und dann an das System **100** über einen B-Kanal des Pfads **9** übertragen werden. Aus der Figur ist auch ersichtlich, dass das Terminal **T1** mit einem herkömmlichen Fernsehmonitor oder einer Videoanzeige **8** assoziiert sein kann. In dieser Weise kann eine Information, die auf der Anzeige **12** des Terminals **T1** angezeigt wird, auch auf dem Monitor **8** angezeigt werden. Um dies zu tun umfasst das Terminal **T1** eine herkömmliche Schaltung zum Umwandeln von VGA-formatierten Signalen in NTSC-Videosignale für eine Anzeige auf dem Monitor B. An sich kann eine Information entweder auf der Terminalanzeige **12** oder dem Monitor **8** oder beiden angezeigt werden.

[0021] Der Benutzer kann auch eine Audioeingabeeinrichtung (oder einen Recorder) **6** als eine Einrichtung zum Übertragen von Audio über das Terminal **T1** an das System **100** für eine Speicherung in dem Speicher **60** als eine Audiodatei verwenden. An sich kann eine Audioeingabeeinrichtung **6** zum Beispiel ein Mikrophon oder ein Audiobandrecorder sein. Audiosignale, die von dem Mikrophon (oder dem Recorder) **6** ausgegeben werden, werden an dem Verstärker **3** angeboten, der dann die Signale an die IDC-Schaltungsanordnung des Terminals **T1** für eine Übertragung an den Pfad **9** sendet, wie voranstehend diskutiert.

[0022] (Es sei darauf hingewiesen, dass der Benutzer auch Daten auf einer Datenbank **61**, die zu dem Server **10** gehört, wie nachstehend diskutiert werden wird, speichern kann).

[0023] In einer illustrativen Ausführungsform der Erfindung wird eine Bibliothek von Multimediafunktionen, die ein einzigartiges Multimediaprotokoll bilden, bereit gestellt, um die Entwicklung und Präsentation einer Multimediaanwendung an einen Benutzer über ein Telekommunikationsnetz zu unterstützen. Insbesondere ist das Protokoll, in Übereinstimmung mit einem Aspekt der Erfindung so angeordnet, dass es unabhängig von der Architektur des Systems **100**, der Multimediaanwendung und dem Terminal **T1** ist. Deshalb können unterschiedliche Multimediafunktionen nahtlos zwischen dem System **100** und dem Terminal **T1** verteilt werden. Das Protokoll ermöglicht somit Kommunikationen zwischen den Elementen, die ein vernetztes Multimediasystem bilden, einschließlich eines Terminals eines Benutzers. In vorteilhafter Weise kann dann eine Multimediaanwendung, die in Übereinstimmung mit dem Protokoll entwickelt ist, auf den meisten Multimediasystemen laufen, vorausgesetzt, dass die Systeme das erfindungsgemäße Protokoll unterstützen.

[0024] Das Protokoll umfasst insbesondere Nachrichten zum Aktivieren von (a) Telekommunikationsfunktionen, (b) Datenzugriffsfunktionen, (c) Audiofunktionen und (d) verteilten Anwendungsfunktionen. Die Telekommunikationsfunktionen umfassen einen Satz von Nachrichten zum Ausbauen einer Kommunikationsverbindung zwischen dem Terminal **T1** und dem System **100** und zum Ausbauen einer Konferenzverbindung. Dieser Nachrichtensatz umfasst (a) das Verbinden, (b) Trennen, (c) Überbrücken eines

Anrufs und (d) das Ausheben einer Überbrückung eines Anrufs. Die Datenzugriffsfunktonen umfassen einen Satz von Nachrichten, umfassend (a) ein Herunterladen einer Datendatei, (b) ein Heraufladen einer Datendatei, (c) das Stoppen einer Datenherunterladung, (d) ein Stoppen einer Datenheraufladung, (e) eine Wiederaufnahme einer Datenherabladung und (f) eine Wiederaufnahme einer Datenheraufladung. Die Audiofunktionen umfassen (a) das Abspielen einer Audiodatei, (b) eine Pause einer Audio-Wiedergabe, (c) eine Wiederaufnahme einer Audio-Wiedergabe, (d) ein Überspringen einer Audio-Wiedergabe, (e) das Einstellen einer Lautstärke, (f) das Stoppen einer Audio-Wiedergabe, (g) das Aufzeichnen von Audio, (h) eine Pause in der Audioaufzeichnung, (i) die Wiederaufnahme einer Audioaufzeichnung und (j) das Stoppen einer Audioaufzeichnung. Die verteilten Anwendungsfunktionen umfassen einen Satz von Nachrichten, umfassend (a) den Start einer Anwendung, (b) das Stoppen einer Anwendung, (c) die Pause einer Anwendung und (d) die Wiederaufnahme einer Anwendung (die verschiedenen Befehle im Zusammenhang mit dem Audioserver sind in dem Anhang A gezeigt). Das Format und die Funktion des Datenservers und die Anwendungscontrollerbefehle sind ähnlich zu denjenigen, die in dem Anhang A gezeigt sind.

[0025] Als Folge des Protokolls und in Übereinstimmung mit einem Aspekt der Erfindung kann eine Multimediaanwendung in logische Komponenten oder Blöcke segmentiert werden, deren Darbietung bzw. Präsentation unter der Steuerung des Benutzers ist. In vorteilhafter Weise werden dann nur diejenigen Blöcke der Multimediaanwendung, die von dem Benutzer angefordert werden, an das Terminal des Benutzers herunter geladen, wobei eine effiziente Verwendung der Telekommunikationsnetzbandbreite durchgeführt werden. In vorteilhafter Weise erlaubt eine derartige Segmentierung, dass die Komponenten der Anwendung in einer Weise gespeichert und übertragen werden, die für den Typ der Daten, die gerade übertragen werden, am besten geeignet ist. Zum Beispiel kann eine Audioinformation in der Datenbank **60** in einem vorgegebenen Format, z. B. einem ADPCM Format gespeichert werden, welches leicht codiert und in ein Sprachsignal für eine Übertragung über das Telefonnetz an den Benutzer umgewandelt werden kann.

[0026] Als Folge des Protokolls kann eine Anwendung in jeweilige Blöcke aufgeteilt (segmentiert) werden. Was dies bedeutet, ist, dass eine Anwendung an einem Benutzer in einer effizienten und schnellen Weise angeboten wird. Zum Beispiel können bestimmte Blöcke einer Anwendung an das Terminal **T1** in einem Hintergrundmodus herunter geladen und darin gespeichert werden, so dass die Blöcke schnell auf die Terminalanzeige **12** herauf gebracht werden können, wenn sie von dem Benutzer aufgerufen werden. Zum Beispiel kann ein Block des Anwendungsprogramms, das eine angezeigte Menueinzelheit mit der höchsten Wahrscheinlichkeit einer Benutzerauswahl unterstützt, zuerst in dem Hintergrund herunter geladen werden, als nächstes kann der Block, der die Menueinzelheit mit der nächst höchsten Wahrscheinlichkeit einer Benutzerauswahl unterstützt, herunter geladen werden usw. Demzufolge ist es wahrscheinlich, dass dann, wenn der Benutzer eine angezeigte Menueinzelheit auswählt, die Anwendungsdaten und/oder die Software, die diese Auswahl unterstützt, in dem Terminal des Benutzers vorhanden (gespeichert) sein wird.

[0027] (Alternativ können die Blöcke sequentiell auf Grundlage ihrer jeweiligen Größen an Stelle ihrer Auswahlwahrscheinlichkeit herunter geladen werden. Dennoch können die Anwendungsblöcke, die die angezeigten Menueinzelheiten unterstützen, parallel und in dem Hintergrundmodus (d. h. mittels eines Vorladens) herunter geladen werden, indem die Netzbandbreite unter den Blöcken auf Grundlage der Wahrscheinlichkeit ihrer zugehörigen Menueinzelheiten, die gerade von dem Benutzer gewählt werden, zugeordnet wird. Als eine andere Alternative kann eine derartige Bandbreitenzuordnung auf die Größe eines Datenblocks gestützt werden. Die Reihenfolge, in der derarige Blöcke sequenziell oder parallel in dem Hintergrund herunter geladen werden, kann auch explizit von dem Anwendungsanbieter spezifiziert werden. Das heißt, ein größerer Abschnitt der Bandbreite wird einem großen Datenblock und nicht einem kleineren Datenblock zugeordnet. Im Gegensatz dazu kann ein größerer Abschnitt der Bandbreite einem kleineren Datenblock zugeordnet werden, um sicher zu stellen, dass der Block auf dem Terminal des Benutzers vorhanden ist, bevor der Benutzer eine Menuauswahl durchführt).

[0028] In Anbetracht der voranstehenden Ausführungen wird nun die Vorgehensweise diskutiert, mit der eine Multimediaanwendung angefordert und an einem Benutzer, der zu der Station S1 und dem Terminal **T1** gehört, dargeboten wird. Wenn der Benutzer einen Befehl zum Aktivieren der Multimediafunktion eingibt, zeigt das Terminal **T1** insbesondere eine Aufforderung für den Benutzer an, um einen einzigartigen Identifizierer des Systems **100** einzugeben, z. B. ein herkömmliches Login, das dem Benutzer zugewiesen ist, und um das Kennwort des Benutzers einzugeben. Als Antwort darauf, dass der Benutzer die Aufforderung erfüllt, veranlasst das Terminal **T1** seine zugehörige Telefonschaltungsanordnung einen ISDN Datenanruf über das Netz **200** an das System **100** zu richten. Wenn das System **100** (der Datenserver **10**) den Anruf annimmt, dann stellt das Netz **200** eine Verbindung zwischen einem der zwei B-Kanäle des Pfads **9** und einem freien B-Kanal des Pfads **11-1** her. Sobald die ISDN-Verbindung aufgebaut ist, überträg dann das Terminal **T1** das Login und Kennwort, das der Benutzer eingegeben hat. Auf einen Empfang davon hin gibt der Server **10** eine Bestätigung an das Terminal **T1** zurück, wenn er feststellt, dass das empfangene Login und das Kennwort gültig sind. Ansons-

ten gibt der Datenserver **10** eine Ungültigkeitsnachricht an das Terminal **T1** zurück und trennt dann den Anruf.

[0029] Zusätzlich zu der Bestätigung gibt der Datenserver **10** ein Menu von Multimediaanwendungen zurück, die auf dem System **100** vorhanden sind, wobei das Menu in einem komprimierten Format ist. Der Datenserver **10** sendet auch die Menuinformation, die identifiziert, wo jede der Multimediaanwendungen, die zu jeweiligen der Menueinzelheiten gehören, in dem Speicher **61** und/oder dem Speicher **60** angeordnet sind. Auf einen Empfang davon dekompriimiert das Terminal **T1** unter Verwendung eines herkömmlichen Dekomprimierungsalgorithmus die Daten (Menu) und zeigt das dekomprimierte Menu auf der Terminalanzeige **12** an. Das Terminal **71** speichert auch die dekomprimierte Anwendungslokalisierungsinformation in seinem internen Speicher.

[0030] Das angezeigte Menu von Multimediaanwendungen kann zum Beispiel ein filmischer Reisebericht, Spiele, Katalogeinkäufe, Erziehung etc. umfassen. Das Menu kann auch eine Multimediaanwendung umfassen, die sich auf das altbekannte Sesamstraßen- (Markenzeichen der Children's Television Workshop Corporation) Fernsehprogramm bezieht. Entsprechend werden die folgenden Einzelheiten im Hinblick auf die letztere Anwendung diskutiert (es sei jedoch darauf hingewiesen, dass die folgende Diskussion gleichermaßen auf die anderen Multimediaanwendungen, z. B. die voranstehend erwähnten Multimediaanwendungen, zutrifft).

[0031] Wenn der Benutzer die Sesamstraßen-Anwendung wählt, beispielsweise durch Zeigen mit einem herkömmlichen Mauszeiger auf die angezeigte Menueinzelheit und durch Betätigen einer zugehörigen Maustaste, dann ermittelt das Terminal **T1** „Anwendungslokalisierungsinformation" in Bezug darauf von seinem internen Speicher und bildet eine Nachricht, die eine Aufforderung zum Übertragen der Sesamstraßen-Anwendung und der Lokalisierungsinformation, die sich darauf bezieht, enthält. Das Terminal **T1** überträgt dann die Nachricht an das System **100** über die ISDN-Verbindung, die von dem Netz **200** aufgebaut wird. Auf eine empfangene Nachricht hin extrahiert der Datenserver **10** die Lokalisierungsinformation von der Nachricht und verwendet diese Information, um den Speicherort des ersten Blocks der angeforderten Anwendung in der Datenbank **61** zu bestimmen. Der Datenserver **10** entlädt den ersten Block der Anwendung und überträgt ihn in seiner komprimierten Form an das Terminal **T1**, wobei der erste Block (a) einen Bildschirm, der auf dem Terminal **T1** angezeigt werden soll, und (b) ein ausführbares Programm umfasst. Auf einen Empfang des ersten Blocks hin dekomprimiert das Terminal **T1** die Daten und führt das empfangene Programm aus. Das letztere Programm zeigt wiederum den begleitenden Schirm auf der Anzeige **12** an und sendet eine Aufforderung an den Audioserver, das Abspielen des Sesamstraßen-Themenlieds zu beginnen. Der Bildschirm zeigt einen Raum, der mit verschiedenen Objekten gefüllt ist, z. B. einem Ball, einem Zug, einem Bücherregal, einem Arkadenspiel etc. Jedes Objekt der Objekte stellt eine Menuauswahl dar. Zum Beispiel wird das Auswählen des (a) Balls bewirken, dass ein springender Ball, begleitet von einem zugehörigen Toneffekt angezeigt wird, (b) das Wählen eines Zugs wird einen Zug anzeigen, der auf einem kreisförmigen Gleis fährt und von einem zugehörigen Toneffekt begleitet wird, oder das Auswählen (c) eines Bücherregals bzw. eines Buchumschlags wird bewirken, dass eine Geschichtenbuchfunktion angezeigt wird. Als ein Aspekt der Erfindung kann das System **100** Teile der Anwendungen, die in Erwägung gezogen werden, herunter laden, während der Benutzer entscheidet, welche Menueinzelheit zu wählen ist, wie voranstehend erwähnt. Zum Beispiel sind der Ball und der Zug Animations-Typ Anwendungen, die vor der Auswahl einer Menueinzelheit durch den Benutzer herunter geladen werden können.

[0032] Wenn der Benutzer zum Beispiel die Zug-Multimediaanwendung (ein sich bewegendes Bild plus Ton) wählt, ist der Anwendungsblock, der diese Auswahl unterstützt, bereits auf dem Terminal **T1** vorhanden. Demzufolge kann das Terminal **T1** schnell auf diese Benutzerauswahl reagieren, indem die Zuganimation angezeigt und der zugehörige Toneffekt abgespielt werden. Die Zuganimation zeigt insbesondere einen sich bewegenden Zug auf einem etwas kreisförmigen Gleis mit einem zugehörigen Audioton an, der eine Zugpfeife charakterisiert. Wenn andererseits der Benutzer die Bücherregal- bzw. Buchumschlag-Menueinzelheit wählt, dann zeigt das Terminal **1** ein Menu an, das diese Anwendung unterstützt, wobei das Menu bereits in einem Terminal **T1** als Folge des Merkmals der vorherigen Herunterladung gespeichert worden ist. Wenn das letztere Menu auf der Anzeige **12** angezeigt wird, zeigt dieses Bilder, die die Buchumschläge von verschiedenen Büchern darstellen, wobei jeder den Titel des jeweiligen Buchs enthält. Das Menu wird von einem Hintergrundton begleitet, d. h. dem Sesamstraßen-Themenlied. Jedes derartige Buch wird aus einer Reihe von Bildern und einem begleitenden Ton gebildet, der eine Geschichte „erzählt". Während der Benutzer entscheidet, welche Menueinzelheit zu wählen ist, lädt das System **100** wie voranstehend beschrieben die erste Seite jedes derartigen Buchs an das Terminal **T1**. Wenn der Benutzer ein Buch wählt, zeigt das Terminal **T1** die vorher gespeicherte erste Seite des gewählten Buchs an. Zusätzlich bildet das Terminal **T1** eine Nachricht, die den Namen der Audiodatei enthält, die zu der gewählten ersten Seite gehört und sendet die Nachricht an das System **100** über den Pfad **9**. Wenn die Anwendung einen Audiokanal erfordert (was der Fall sein würde, um den voranstehend erwähnten Toneffekt abzuspielen), dann sendet die Anwendung eine Aufforderung diesbezüglich an den Audioserver **30** über den Datenserver **10**. Der Audioserver **30**, der auf die Aufforderung anspricht,

sendet die geeigneten Befehle an das Netz **200**, um den Pfad **11-2** herzustellen.

[0033] Auf eine empfangene Nachricht hin übergibt der Datenserver **10** die Aufforderung an den Audioserver **30** über das LAN **45**. Der Server **30** wiederum entlädt die identifizierte Datei von dem Speicher **60** und überträgt die Datei als einen kontinuierlichen (isochronen) Bitstrom über einen zugehörigen B-Kanal des Pfad **11-2**, der sich an das Terminal **T1** über das Netz **200** erstreckt. Das Terminal **T1** wandelt den Bitstrom wiederum, wenn er empfangen wird, in analoge Signale für eine Darbietung an der Station S1 oder dem Audiosystem **5** um. Während der Benutzer dem Audioton zuhört und die Seite betrachtet, lädt das System **100** in ähnlicher Weise die nächste Seite des gewählten Buchs an das Terminal **T1** herunter. Das Terminal **T1** und das System **100** setzen einen Betrieb in der voranstehend beschriebenen Weise fort, bis die letzte Seite des gewählten Buchs dem Benutzer dargeboten worden ist. An diesem Punkt kann der Benutzer eine erneute Anzeige des Buchmenus anfordern und wie voranstehend diskutiert vorgehen. Alternativ, wenn das Buchauswahlmenu angezeigt wird, kann der Benutzer das Hauptmenu auswählen.

[0034] Als ein Aspekt der Erfindung kann ein Benutzer, z. B. der Benutzer, der zu der Station S1 und dem Terminal **T1** gehört, auf einen „live" Agenten, z. B. ein Reisebüro zugreifen, während eine bestimmte Multimediaanwendung betrachtet wird. Wenn der Benutzer z. B. eine Multimedia-Fihnreiseberichtanwendung auf der Anzeige **12** mit einem zugehörigen Ton, der entweder von der Station S1 oder dem Audiosystem **5** ausgegeben wird, betrachtet und der Benutzer den Wunsch hat, mit einem Reisebüro verbunden zu werden, dann kann der Benutzer eine Aufforderung für eine derartige Verbindung an dem System **100** eingeben. Der Benutzer kann dies tun, indem er den Schirmcursor (Schirmbildmarke) an ein bestimmtes Icon richtet, das mit dem Filmreisebericht angezeigt wird, und indem er das Icon in einer herkömmlichen Weise wählt. Das Terminal **T1** sendet im Ansprechen darauf eine Anruf_Überbrücken-Nachricht (Bridge_Call Nachricht, wie nachstehend diskutiert), an das System **100**. Auf einen Empfang dieser Nachricht hin weist der Datenserver **100** den Audioserver **30** über das LAN **45** an, einen Anruf über einen verfügbaren ISDN B-Kanal des Pfads **11-2** an einen Agenten (Betreiber), der zu der Anwendung gehört, beispielsweise eine Hilfsperson oder einen Operator **225**, zu richten. Wenn der Anruf beantwortet wird, signalisiert der Audioserver **30** dann dem Netz **200**, diesen Anruf zu dem Anruf des Benutzers und hin zu dem System **100** erstreckend über den Pfad zu 9 zu überbrücken. Der Audioserver **30** trennt sich dann von dem überbrückten Anruf und die Filmreiseberichtanwendung wird unterbrochen, bis der Anruf zu dem Agenten fertig gestellt ist. Wenn der Anruf fertig gestellt ist, dann führt der Audioserver **30** eine Neuverbindung zu dem Terminal **T1** über den Pfad **11-2** aus. Das System **100** setzt dann die Darbietung der Filmreiseberichtanwendung fort.

[0035] Wie sich der **Fig.** 1 entnehmen lässt, umfasst das System **100** auch eine Arbeitsstation **40** und eine Anwendungsentwicklungsplattform (Application Development Platform, ADP) **150**, die verwendet werden, um das System **100** mit dem Multimediaanwendungs-Entwicklungsgerät, das zu Entwicklern von Multimediaanwendungen gehört, beispielsweise Anwendungsanbietern **125** bzw. **175**, zu koppeln. In einer illustrativen Ausführungsform der Erfindung stellt die Arbeitsstation (Work Station) **40**, die zum Beispiel der Gateway **200** Computer sein kann, der von Gateway Inc. erhältlich ist, einen Mechanismus bereit, mit den ein Anwendungsanbieter, z. B. ein Anbieter **125**, eine Anwendung speichern kann, die gerade in dem System **100** (Datenbanken **60** und **61**) entwickelt wird. Der Anbieter **125** kann dies tun, indem er einen ISDN Anruf über den BRI ISDN Pfad **126** zum Herstellen einer Verbindung zu der Arbeitsstation **40** richtet. Sobald die Verbindung hergestellt worden ist, kann der Anbieter **125** dann mit der Arbeitsstation **40** kommunizieren, um die letztere anzuweisen, eine Anwendung in dem Speicher **60** und/oder **61** zu speichern. Um den Transfer der Anwendung an das System **100** zu bewirken, überträgt der Anbieter **125** unter Verwendung des Heraufladedatenbefehls (nachstehend diskutiert) jede Datei, die die Anwendung bildet, an die Arbeitsstation **40** über den Pfad **126**. Die Arbeitsstation **40** speichert im Ansprechen auf einen Empfang einer Datei die Datei in dem Speicher **60** oder **61** auf Grundlage des Typs von Daten (Audiodaten oder digitale Daten) in der empfangenen Datei. Sobald sämtliche derartigen Dateien in dem System **100** gespeichert worden sind, kann der Anbieter **125** dann die gespeicherte Anwendung „debuggen", indem ein Anruf an die Anwendungsentwicklungsplattform (ADP) **150** über den Pfad **126** durchgeführt wird.

[0036] Die ADP **150**, die ebenfalls ein Gateway **2000** Computer sein kann, repliziert die Server **10** und **30** und den Controller **20**, so dass sie den Anruf beantwortet und dann die Funktionen des Systems **100** im Zusammenhang mit einer Darbietung der gespeicherten Anwendung des Anbieters an dem Anbieter, als ob der Anbieter ein Benutzer wäre, beispielsweise der Benutzer, der zu der Station S1 und dem Terminal **T1** gehört, emuliert. In Übereinstimmung mit einem Aspekt der Erfindung stellt das System **100** somit eine Funktionalität bereit, die einem Anwendungsanbieter erlaubt, in dem System **100** eine Anwendung zu speichern, die der Anbieter gerade entwickelt und dann über das Telekommunikationsnetz **200** auf das System **100** zuzugreifen, um die Anwendung zu „debuggen" (d. h. Fehler zu entdecken und zu korrigieren). Somit wird der Anwendungsanbieter (Provider) mit einer zugehörigen Anwendung im Grunde genommen in der gleichen Weise, wie ein Benutzer schließlich mit der Anwendung gekoppelt sein wird, gekoppelt.

[0037] Die Software, die eine Multimediaanwen-

dung steuert, kann zwischen einem Terminal **T1** und dem System **100** aufgeteilt sein. Es wird nun auf die **Fig.** 2 Bezug genommen. Dort ist das Hauptmenuprogramm gezeigt, das in einem Terminal **T1** gespeichert ist und das einem Benutzer, der an dem Terminal **T1** arbeitet, erlaubt, auf ein Multimediasystem oder die Plattform 100 zuzugreifen. Insbesondere dann, wenn der Benutzer das Hauptmenuprogramm aktiviert, wird in das Programm im Block **300** übergegangen und die Ausführung geht zum Schritt 301, wo sie den Benutzer auffordert, sich einzuloggen und ein Kennwort einzugeben. Wenn der Benutzer dies tut, geht das Programm zum Block **302**, wo es die voranstehend erwähnte ISDN Platine sowie die Software, die den Betrieb der Platine steuert, initialisiert. Das Programm stellt dann einen Telefonanruf an der Multimediaplattform **100** unter Verwendung eines Kanals der zwei Basisraten-ISDN-Kanäle her (Block **303**). Wenn der Telefonanruf aufgebaut ist, dann kommunizieren das Hauptmenuprogramm und das System (die Plattform) **100** miteinander. Derartige Kommunikationen umfassen das Übergeben des eingegebenen Kennworts und des Logons an die Plattform 100. Wenn die Plattform **100** feststellt, dass das Logon ungültig ist, beendet die Plattform **100** dann die ISDN-Verbindung. Das Programm zeigt im Ansprechen darauf eine Nachricht an, die diese Tatsache anzeigt und endet dann. Wenn jedoch die Plattform feststellt, dass das eingegebene Logon gültig ist, dann informiert sie das Programm über die Tatsache (Block **304**) und lädt ein Menu von Multimediadiensten, auf die der Benutzer zugreifen möchte, herunter. Auf einen Empfang davon zeigt das Programm derartige Dienste an (Block **305**) und wartet auf den Benutzer, dass er eine Auswahl eingibt (Block **306**). Wenn der Benutzer eine Auswahl eingibt, sendet das Programm (Block **306**) die Auswahl an die Plattform 100 und lässt dann die Auswahl ablaufen (Block **307**), wenn sie von der Plattform **100** empfangen wird, wie nachstehend diskutiert. Wenn die Multimediaauswahl abgeschlossen ist, dann zeigt das Programm das Menu der Dienste erneut an (Block **308**) und wartet auf den Benutzer, dass er eine neue Auswahl eingibt. Wenn der Benutzer eine Auswahl eingibt, dann sendet das Programm die eingegebene Auswahl (Block **306**) an die Multimediaplattform, wie voranstehend diskutiert. Wenn andererseits der Benutzer wünscht, die Session zu beenden und einen Anzeiger eingibt, der diese Tatsache anzeigt, wenn z. B. ein „q" für Aufhören (quit), dann informiert das Programm die Plattform darüber und beendet die ISDN-Verbindung.

[0038] Das Hauptmenuprogramm wird von einer Anzahl von anderen Programmen oder Aufgaben (Tasks) unterstützt, die dafür ausgelegt sind, um in dem Hintergrundmodus zu laufen. Ein derartiges Programm führt eine ISDN Task aus, wie in **Fig.** 3 gezeigt. Insbesondere dann, wenn im Block **400** in die Task übergegangen wird, schreitet sie fort, eine Anzahl von Funktionen zu initialisieren (Block **401**). Eine derartige Initialisierung umfasst das Aufbauen von Warteschlagen, die für die Speicherung von. Datendateien benötigt werden, die sich auf die gewählte Anwendung beziehen. Derartige Datendateien umfassen das Herunterladen von Dateien von der Plattform an das Terminal des Benutzers und umgekehrt. Die Initialisierung umfasst auch die Aktivierung von Datenkomprimierungs- und Dekomprimierungs-Tasks. Das Programm wartet dann auf den Empfang einer Aufforderung von der Anwendung, d. h. einer Aufforderung zum „Holen" („get") einer nächsten Datendatei (Block **402**). Die ISDN-Task umfasst auch die Aktivierung (Block **403**) einer sogenannten Buchführungs-Task (Block **408**), die die Einlog- und Auslog-Funktionen verfolgt, die das Terminal des Benutzers während seiner Kommunikationen mit der Plattform ausführt, sowie andere ISDN-Funktionen, wie nachstehend beschrieben wird. Die ISDN-Task (Block **405**) aktiviert auch eine Datentransferfunktion, die arbeitet, um digitale Datendateien zwischen dem Terminal des Benutzen und der Plattform zu transferieren (Block **409**). In ähnlicher Weise aktiviert die ISDN-Task auch (Block **406**) eine Audiofunktion (Block **410**), die unter der Richtung der gewählten Anwendung arbeitet, um eine digitalisierte Audiodatei (digitalisierte Audiodateien) von der Anwendungsplattform zu empfangen und den Audioton über den Lautsprecher **4** „abzuspielen". Wenn die Aufforderung, die von der gewählten Anwendung empfangen wird, keine der obigen Funktionen ist (Blöcke **403**, **405** oder **406**), dann sendet die ISDN-Task (Block **407**) eine sogenannte negative Bestätigung an die Anwendung und wartet auf den Empfang der nächsten Aufforderung von der gewählten Anwendung (Block **402**).

[0039] Ein Flussdiagramm der voranstehend erwähnten Buchführungs(Bookkeeping)-Funktion 403 ist in **Fig.** 4 dargestellt. Insbesondere dann, wenn die Buchführungsfunktion aktiviert wird (Block **500**) bestimmt sie, ob die empfangene Aufforderung eine Aufforderung zum Initialisieren der voranstehend erwähnten ISDN-Hardware ist (Block **501**). Wenn dies der Fall ist, dann initialisiert das Programm die ISDN-Hardware (Block **506**). Die Aufforderung, eine derartige Initialisierung durchzuführen, ergibt sich gewöhnlicherweise, wenn das Anwendungsprogramm folgert, dass der ISDN-Kommunikationspfad zwischen dem Terminal des Benutzers und der Plattform **100** in einem nicht bekannten Zustand ist, z. B. sich „aufgehängt" hat. Wenn dies nicht der Fall ist, dann prüft das Programm (Block **502** /**503**), um nachzusehen, ob die Aufforderung eine Aufforderung für einen bestimmten Typ von ISDN-Verbindung/– Trennung ist. Für eine Verbindungsaufforderung sendet das Programm (Block **507**) einen sogenannten Verbindungsbefehl, der Parameter enthält, die die bestimmte Verbindung definieren, an die Plattform **100**. Für eine Trennungsaufforderung sendet das Programm (Block **508**) einen Trennungsbefehl, der die Parameter enthält, die die Trennung definieren, an die Platt-

form **100**. Wenn die bestimmte Funktion (d. h. Block **506**, **507** oder **508**) erfolgreich ausgeführt wird (Block **504**), dann setzt das Programm (Block **505**) einen Anzeiger, der „VAL" bezeichnet wird, auf eine Eins, um anzuzeigen, dass die angeforderte Funktion erfolgreich abgeschlossen wurde, und kehrt dann zum Block **402** zurück. Ansonsten setzt das Programm VAL auf Null (Block **509**), um anzuzeigen, dass die angeforderte Funktion **405** fehlgeschlagen ist.

[0040] Ein Flussdiagramm der voranstehend erwähnten Datenfunktion ist in **Fig.** 5 dargestellt. Wenn die Datenfunktion aktiviert wird (Block **600**), bestimmt sie insbesondere den Typ der Datenanforderung, die von der gewählten Anwendung empfangen wurde. Wenn festgestellt wird, dass die Datenanforderung eine sogenannte Anforderung zum Herunterladen von Daten von der Plattform an das Terminal des Benutzers ist (Block **601**), dann fügt das Programm die Aufforderung (Block **604**) zu einer sogenannten Herunterladungsprozess-Warteschlange hinzu und kehrt dann zum Block **402** der **Fig.** 3 zurück. Wenn andererseits festgestellt wird, dass die Aufforderung eine Aufforderung zum Heraufladen von dem Terminal des Benutzers an die Plattform ist (Block **602**), dann speichert das Programm die Aufforderung (Block **605**) in einer sogenannten Herunterladungsprozess-Warteschlange und kehrt dann zum Block **402** zurück. Wenn das Programm feststellt, dass die Aufforderung irgendetwas anderes als eine Herunterlade- oder Heraufladeaufforderung ist (Block **603**), wird eine bestimmte Aufforderung, die bereits für eine Verarbeitung in eine Warteschlange eingereiht worden ist, gelöscht, und dann aktualisiert das Programm den Status (Block **606**) für die bestimmte Aufforderung. Das Programm (Block **607**) sendet eine Löschungsnachricht in Bezug auf die bestimmte Aufforderung an die Plattform, d. h. den Daten- oder Audioserver. Das Programm kehrt dann zum Block **402** zurück.

[0041] **Fig.** 6 zeigt in einer Flussdiagrammform die Task oder das Programm, das in dem Hintergrundmodus abläuft, um eine Datendatei-Herunterladefunktion auszuführen. Wenn es aktiviert ist (Block **700**) prüft das Programm (Block **701**), um nachzusehen, ob es eine Aufforderung zum Herunterladen einer Datei von der Plattform an das Benutzerterminal empfangen hat. Wenn dies der Fall ist, dann speichert das Programm (Block **703**) die Aufforderung in einer zugehörigen Warteschlange und geht dann zum Block **702**. Das Programm (Block **702**) prüft dann, um nachzusehen, ob es gegenwärtig Daten von der Plattform empfängt und kehrt zu dem Block **701** zurück wenn es feststellt, dass dies der Fall ist. Wenn nicht, dann prüft es (Block **704**), um nachzusehen, ob seine zugehörige Warteschlange leer ist und kehrt zum Block **701** zurück, wenn dem so ist. Ansonsten aktiviert (erzeugt) das Programm (Block **705**) eine Task, die dafür ausgelegt ist, um die Datendatei, wenn sie von der Plattorm empfangen wird, (a) zu lesen und zu dekomprimieren und (b) die Datei in Übereinstimmung mit Befehlen, die in der Datei enthalten sind, zu verarbeiten. Das Programm kehrt dann zum Block **701** zurück.

[0042] **Fig.** 7 zeigt in einer Flussdiagrammform die Task oder das Programm, das in dem Hintergrundmodus läuft, um eine Datendatei-Heraufladefunktion auszuführen. Es ist ersichtlich, dass das in **Fig.** 7 gezeigte Programm etwas ähnlich zu dem in **Fig.** 6 gezeigten Programm ist, aber eine andere Funktion ausführt. Wenn das Programm aktiviert wird (Block **800**), prüft das Programm (Block **801**) insbesondere, um nachzusehen, ob es eine Aufforderung zum Übertragen einer bestimmten Datei von dem Benutzerterminal an die Plattform, d. h. den Daten- oder Audioserver, empfangen hat. Wenn dies der Fall ist, dann speichert das Programm (Block **803**) die Aufforderung in einer zugehörigen Warteschlange und geht dann zum Block **802**. Das Programm (Block **802**) prüft, um nachzusehen, ob es gegenwärtig Daten an die Plattform sendet und kehrt zum Block **801** zurück, wenn es feststellt, dass dies der Fall ist. Wenn das Programm feststellt, dass es sich nicht in einem Heraufladem odus befindet, dann prüft es (Block **804**), um nachzusehen, ob seine zugehörige Warteschlange leer ist und kehrt zum Block **801** zurück, wenn es feststellt, dass dies der Fall ist. Ansonsten aktiviert (erzeugt) das Programm (Block **805**) eine Task, die die Datendatei, die herauf geladen werden soll, komprimiert und überträgt die Datei an die Plattform. Das Programm kehrt dann zum Block **701** zurück.

[0043] Die **Fig.** 8 bis 19 zeigen in einer Flussdiagrammform die Programme, die die Erfindung in dem Audioserver **30** (**Fig.** 1) implementiern. Insbesondere und Bezug nehmend auf **Fig.** 8 wird dann, wenn der analoge Server eingeschaltet und „herauf gefahren" wird, das Programm im Block **900** begonnen, wo es fortschreitet (Block **901**), um eine InterFace- (I/F, Schnittstellen) Schaltungsplatine, die den Audioserver mit dem Netz koppelt, zu initialisieren. Das Programm initialisiert auch eine Anzahl von Programmvariablen und wartet dann (Block **902**) auf einen Empfang einer Nachricht von dem Datenserver, der anfordert, dass ein Audio-ISDN-B-Kanal durch das Netz **200** mit dem Benutzerterminal verbunden wird. Im Ansprechen auf einen Empfang der Nachricht erzeugt das Programm (Block **903**) einen B_Kanal_Serverprozess (B_Channel_Serverprocess) und ordnet den letzteren Prozess dem B_Kanal (B_Channel) zu, der mit dem Benutzerterminal verbunden ist, wie nachstehend beschrieben. Das Programm (**904**) speichert dann die empfangene Nachricht in der Warteschlange, die zu dem neu erzeugten Prozess gehört und kehrt dann zum Block **902** zurück, um auf einen Empfang der nächsten Verbindungsnachricht von dem Datenserver **10** zu warten.

[0044] **Fig.** 9 zeigt in einer Flussdiagrammform den B_Kanal_Serverprozess, der erzeugt wird, um einen bestimmten ISDN B-Kanal zu bedienen, der mit dem Audioserver verbunden ist. Das heißt, ein B_Kanal

Serverprozess wird erzeugt, um jede derartige Verbindung zu bedienen. Insbesondere dann, wenn der B_Kanal_Serverprozess („Programm") erzeugt ist, wird dieser Prozess im Block **1000** begonnen, wo er eine Anzahl von Programmvariablen initialisiert, die er zum Bedienen eines bestimmten ISDN B-Kanals verwendet. Das Programm (Block **1001**) entlädt eine Aufforderungsnachricht von seiner zugehörigen Warteschlange und verzweigt sich dann (Block **1002**) auf den Typ von Aufforderung hin, der in der Nachricht enthalten ist. Eine derartige Verzweigung veranlasst das Programm, sich an eines von 14 Unterprogrammen **1003** bis **1016** zu verzweigen, die in der Figur dargestellt sind. Das Programm verzweigt sich an das letzte derartige Unterprogramm **1017**, wenn die Aufforderungsnachricht von einem unbekannten Typ ist. Das Unterprogramm **1017** zeichnet den Fehler auf und kehrt dann zum Block **1001** zurück, um die nächste Aufforderungsnachricht zu verarbeiten, die in der zugehörigen Warteschlange gespeichert ist.

[0045] Die verbleibenden Unterprogramme **1003** bis **1016** werden nun wiederum diskutiert. Insbesondere ist eine ausgedehnte Version des Blocks **1003** in **Fig.** 10 dargestellt. Wenn das Programm den Block **1003** erreicht, bedeutet dies, dass die Aufforderungsnachricht eine Aufforderung zum Herstellen einer B-Kanal Audioverbindung einer Primärrate (PRImary (PRI) Rate) zwischen dem Audioserver und dem Benutzerterminal ist. Um dies zu tun, überprüft das Programm (Block **1003-2**) zunächst eine sogenannte globale Informationstabelle, auf die sowohl von dem Daten- als auch von dem analogen Server zugegriffen wird, um zu bestimmen, ob die zwei B-Kanä1e, die einem Benutzer zugewiesen werden können, bereits als Datenpfade durch den Datenserver **10** und/oder den Audioserver **30** zugeordnet worden sind. Wenn dies der Fall ist, dann kann das Programm auf die Verbindungsnachricht nicht antworten. Andererseits führt das Programm eine Zuweisung (Zuordnung) eines freien B-Kanals zu der Verbindung aus und aktualisiert dann eine zugehörige Speichertabelle, um zu zeigen, dass der zugewiesene Kanal in Verwendung ist. Das Programm (Block **1003-3**) sendet dann einen Befehl zum Aufbauen bzw. Einrichten der B-Kanal-Verbindung zu der Inter-Face (I/F) Schaltungsplatine, die den Audioserver mit dem ISDN-Netz koppelt (wie voranstehend erwähnt), wobei der Befehl die angerufene Telefonnummer und die Kanalnummer des gewählten B-Kanals enthält. Die I/F-Schaltung sendet wiederum eine Anrufaufbaunachricht, die die Telefonnummer und eine Kanalnummer über den zugeordneten ISDN-D-Kanal enthält, wodurch ein Anruf an das Benutzerterminal effektiv gerichtet wird. Das Programm (Block **1003-4**) aktualisiert dann die voranstehend erwähnte globale Informationstabelle, die verwendet wird, um den Status der zwei ISDN-Kanäle, die jedem aktiven Benutzer zugeordnet werden können, und der Anwendung, die von dem jeweiligen Benutzer gewählt wird, nach zu verfolgen. Das Programm kehrt dann zum Block **1001** zurück.

[0046] Eine erweiterte Version des Blocks **1004** ist in **Fig.** 11 dargestellt, die begonnen wird (Block **1004-1**), wenn sich die empfangene Aufforderung auf eine Aufforderung zum Abspielen einer bestimmten analogen Datei (von bestimmten analogen Dateien) bezieht. Wenn das Programm begonnen wird, prüft das Programm (Block **1004-2**), um nachzusehen, ob ein Abspielserver bereits als Folge einer ähnlichen, aber vorangehenden Aufforderung bereits aktiviert (erzeugt) worden ist. Wenn dies der Fall ist, dann speichert das Programm (Block **1004-3**) die gegenwärtige Aufforderung in der Warteschlange des Wiedergabeservers, der erzeugt worden ist, um die Aufforderung des Benutzers für eine Audioanwendung, z. B. Wiedergabe einer Musikdatei über den verbundenen B-Kanal, zu behandeln bzw. zu bedienen. Das Programm kehrt dann zum Block **1001** zurück. Wenn dies nicht der Fall ist, dann erzeugt bzw. initiiert das Programm (Block **1004-5**) eine Wiedergabe-Serverroutine (playbk_server), um die Benutzeraufforderung zu behandeln und geht dann zum Block **1004-3** weiter.

[0047] Das Programm das im Block **1004-4** (**Fig.** 11) aktiviert bzw. initialisiert worden ist, ist in **Fig.** 12 gezeigt. Wenn das Programm begonnen wird, initialisiert das Programm (Block **1004-41**) insbesondere eine Anzahl von Variablen und Listen, z. B. eine Wiedergabeliste, die es während der Zeit, in der es läuft, verwendet. Das Programm (Block **1004-42**) prüft dann seine Warteschlage, um nachzusehen, ob sie eine oder mehrere Abspielaufforderungen von Dateien enthält, die bereit sind, um an den Benutzer zurück „abgespielt" zu werden. Wenn die Warteschlange leer ist, dann endet das Programm (das heißt, das Programm führt einen Ablauf fort, so lange wie ein Eintrag in der zugehörigen Warteschlange enthalten ist). Ansonsten baut das Programm (Block **1004-43**) eine Liste von Dateien, z. B. Dateien, die digitalisierte Musik oder eine Geschichte enthalten, die einem Anzeiger zugeordnet sind, der anzeigt, dass die Datei „abgespielt" werden kann. Insbesondere bedeutet das „Abspielen einer Datei", dass das Programm die digitalisierten Daten, die die Datei bilden, von ihrem zugehörigen Audiospeicher **60** entlädt und die Daten an die I/F-Platine liefert. Die I/F-Platine wandelt wiederum die digitalisierten Daten in eine analoge Form um und überträgt das Ergebnis über den analogen B-Kanal, der dem Benutzer zugewiesen ist. Was mit einem Anzeiger gemeint ist, ist eine Aufforderung zum Pausieren, Überspringen oder Wiederaufnehmen (Blöcke **1005**, **1007** bzw. 1006) des Abspielvorgangs der Datei. Eine Aufforderung, die eine Datei identifiziert, aber zu einer anderen in der Warteschlange eingereihten Aufforderung (einem Anzeiger) zum Pausieren der Wiedergabe der Datei gehört, wird von der Warteschlange nicht entfernt und bleibt in der Warteschlange, bis eine Aufforderung zur Wiederaufnahme der Wiedergabe der Datei empfangen und in der zugehörigen Warte-

schlange gespeichert wird. Eine Aufforderung zum Stoppen (Block **107**) der Wiedergabe einer Datei oder zur Beendigung der „Wiedergabe" der Inhalte einer Datei veranlasst das Programm eine derartige Wiedergabe zu stoppen und die zugehörige Aufforderung, die diese Datei identifizier, zu löschen.

[0048] Wenn das Programm das Bauen einer derartigen Liste fertig gestellt hat, geht es dann (Block **1004**-**45**) (Block **451**, FIG. 13) in einen Prozess Wiedergabe_Liste (Play_List) über, der in der voranstehend beschriebenen Weise bewirkt, dass jede Datei, die in der Liste enthalten ist, abgespielt wird, d. h. an den Benutzer übertragen wird. Die tatsächliche Wiedergabe bzw. das Abspielen der Datei wird durch einen Datei_Wiedergabe-Prozess (File_Player Process) durchgeführt, der im Block **454** initiiert wird. Wenn jedoch der Datei_Abspiel Prozess bereits aktiv ist (Block **452**), dann fügt das Programm die Aufforderung zu der Warteschlange der Datei Abspielgerät hinzu. Wenn, zurückkehrend zur **Fig.** 12, die Wiedergabeliste leer ist (Block **1004-44**), dann wartet das Programm (Block **1004-46**), bis ein neuer Eintrag (z. B. eine Wiederaufnahme- oder Wiedergabe-Aufforderung) in der zugehörigen Aufforderungswarteschlange gespeichert worden ist. Auf den Empfang einer derartigen Benachrichtigung hin kehrt das Programm zum Block **1004-42** zurück.

[0049] Der Datei_Wiedergabe-Prozess (File_Player Process), der im Block **454** aktiviert wird, ist in **Fig.** 14 gezeigt und beginnt am Block **454-1**. Insbesondere initialisiert das Programm (Block **454-1**) verschiedene Softwarevariablen und bestimmt dann (Block **454-2**), ob seine zugehörige Warteschlange leer ist. Wenn dies so ist, dann endet das Programm. Ansonsten aktualisiert das Programm die Informationstabelle des Benutzers, um darauf hinzuweisen, dass der Benutzer das Abspielen einer Anzahl von Dateien (d. h. einen Wiedergabemodus) angefordert hat. Das Programm „spielt" dann die Datei in der voranstehend diskutierten Weise ab, d. h. durch Entladen der Inhalte der Datei und durch Übergeben von diesen an die I/F-Platine. Während einer derartigen Entladung führt das Programm eine Nachverfolgung aus, um nachzusehen, ob es einen Dateiende- (End Of File, EOF)-Anzeiger antrifft oder eine Benutzeraufforderung zum Stoppen einer Wiedergabe der Datei empfangen hat. Wenn das Programm (Block **454-5**) eine Benutzeraufforderung (RQST) zum Stoppen einer Wiedergabe der Datei antrifft, dann sendet er (Block **454-6**) einen Befehl an die I/F-Platine, um die Übertragung der Datei zu beenden, was zu der Beendigung der Wiedergabe dieser Datei führt. Ansonsten kehrt das Programm zum Block **454-2** zurück.

[0050] Wie voranstehend diskutiert, ermöglicht die Vielseitigkeit der Multimediaplattform **100** einem Anwendungsentwickler die Option, dass einem Benutzer erlaubt wird, eine analoge Datei, beispielsweise Musik, in der Diskdatei **60** zu speichern. Diesbezüglich kann der Benutzer die Option durch Eingeben einer Aufforderung zum Speichern einer Audiodatei in der Disk **60** aktivieren, wobei die Aufforderung an den B-Kanal Server über die Daten- und Audioserver geliefert wird (die Idee zum Speichern einer Datei, die von dem Benutzer in dem Diskspeicher **60** stammt, wird hier auch als „Aufzeichnen" der Datei bezeichnet). Im Ansprechen auf den Empfang der Aufforderung hin verzweigt der B-Kanal Server sich zu dem Block **1009**, um die Aufforderung zu verarbeiten. Eine ausgedehnte Version des Blocks **1009** ist in **Fig.** 15 dargestellt. Wenn sie begonnen wird (Block **1009-1**), prüft die Aufforderung Aufzeichnen (Record_Request) Routine (Block **1009-2**), um nachzusehen, ob ein Aufzeichnungsserver bereits aktiviert worden ist, um die Aufforderung des Benutzers zu behandeln. Wenn dies der Fall ist, dann speichert das Programm (Block **1009-3**) die Aufforderung des Benutzers in einer Warteschlange, die zu dem Aufzeichnungs Server-Prozess gehört. Das Programm speichert auch in der Warteschlange die Identität (den Ort) der Datei, die verwendet werden wird, um die Aufzeichnung des Benutzers zu speichern. Das Programm kehrt dann zum Block **1009-1** zurück. Wenn ein Server nicht so aktiviert bzw. erzeugt worden ist, dann erzeugt das Programm (**1009-4**) eine Kopie des Servers, um die Aufforderung zu behandeln und geht dann zum Block **1009-3**.

[0051] **Fig.** 16 zeigt in einer Flussdiagrammform den Aufzeichen_Server-Prozess, der im Block **1009-4** der **Fig.** 15 erzeugt wird und der dafür ausgelegt ist, um analoge Daten, die z. B. Musik charakterisieren, in der Disk **60** für eine zukünftige Verwendung von dem Benutzer zu speichern. Insbesondere dann, wenn das Programm gestartet wird, (Block **1009-41**), initialisiert das Programm eine Anzahl von Variablen und Listen, z. B. eine Aufzeichnungsliste, die es während seines gegenwärtigen Eintrags verwendet. Das Programm (Block **1009-42**, **1009-43**, **1009-44** und **1009-45**) prüft dann seine Warteschlange, um nachzusehen, ob die Warteschlange eine oder mehrere Aufzeichnungsaufforderungen von Dateien enthält, die bereit sind, um in dem Speicher für den Benutzer gespeichert zu werden. Wenn die Warteschlange leer ist, dann endet das Programm (das heißt, das Programm setzt einen Ablauf so lange fort, wie ein Eintrag in seiner zugehörigen Warteschlange enthalten ist). Ansonsten baut das Programm (Block **1009-43**) eine Liste der Dateien, die zu einem Anzeiger gehören, und von denen das Programm erwartet, dass es sie über die I/F-Platine für eine Speicherung in dem Speicher **60** empfängt. Insbesondere bedeutet das „Aufzeichnen einer Datei", dass das Programm die Daten, die die Datei bilden, von der I/F-Platine, die zu dem B-Kanal gehört, der mit dem Benutzer verbunden ist, empfangen und die Daten im Speicher **60** für den Benutzer speichern wird. Die I/F-Platine wandelt auf einen Empfang der analogen Daten über den betreffenden B-Kanal hin die analogen Daten in eine digitale Form um. Der Aufzeichnungs_Server (d. h. der Aufzeichnen_Liste-Prozess, der nachstehend disku-

tiert wird) speichert dann die digitalen Daten, wie voranstehend diskutiert. Was mit einem Anzeiger gemeint ist, ist eine Aufforderung zum (a) „Aufzeichnen" der Datei, (b) „Pausieren der Aufzeichnung der Datei" (Block **1010**), (c) „Wiederaufnehmen" der Aufzeichnung der Datei" (Block **1011**) und (d) zum Stoppen der Aufzeichnung der Datei (Block **1012**). Eine Aufforderung, die eine Datei identifiziert, aber zu einer in einer Warteschlange eingereihten Aufforderung (einem Anzeiger) gehört, um das Aufzeichnen der Datei zu pausieren, wird von der Warteschlange nicht entfernt und verbleibt in der Warteschlange, bis eine Aufforderung zur Wiederaufnahme einer derartigen Aufzeichnung empfangen und in der zugehörigen Warteschlange gespeichert wird. Eine Aufforderung zum Stoppen der Aufzeichnung einer Datei oder auf einem Empfang der letzten Daten, die die Datei bilden, veranlasst das Programm, eine derartige Aufzeichnung zu stoppen. Ein Aspekt des Programms (Block **1009-46**) ist der Eintritt in einen Aufzeichnungs-Liste-Prozess, der in **Fig.** 17 dargestellt ist.

[0052] Es sei darauf hingewiesen, dass die Struktur des Aufzeichnungs-Liste-Prozesses ähnlich zu dem Aufbau der Wiedergabe_Liste-Prozess ist, der in **Fig.** 13 gezeigt ist. An sich wird **Fig.** 17 hier nicht diskutiert, da der Betrieb des Wiedergabe_Liste-Prozesses einfach durch Bezugnahme auf sowohl die **Fig.** 13 als auch die **Fig.** 17 einfach verstanden werden kann. Jedoch reicht es aus, um zu sagen, dass der Aufzeichnen_Liste-Prozess (a) einen Datei Recorder-Prozess aktiviert, wenn er noch nicht bereits aktiv ist und den Benutzer bedient, und (b) die Aufzeichnungsliste zu der Warteschlange des Datei-Recorder-Prozesses hinzufügt und dann zum Block **1009-42** zurückkehrt.

[0053] **Fig.** 18 zeigt in einer Flussdiagrammform den Datei-Recorder-Prozess, der von dem Aufzeichnen_Liste-Prozess aktiviert bzw. hervorgebracht wird. Aus **Fig.** 18 lässt sich ersehen, dass der Aufbau des Datei Recorder-Prozesses irgendwie ähnlich zu demjenigen des Datei_Abspielgerät-Prozesses ist. Demzufolge wird **Fig.** 18 hier nicht diskutiert, da die Funktionen, die von den verschiedenen Blöcken ausgeführt werden, die der Datei_Recorder-Prozess umfasst, leicht als Ergebnis der Ähnlichkeit und der voranstehenden Diskussion des Datei_Abspielgerät-Prozesses ermittelt werden können.

[0054] Für den Fall, dass der Benutzer eine Datei „abhört" und die Wiedergabelautstärke zu hoch oder zu niedrig ist, dann kann in diesem Fall der Benutzer eine Aufforderung eingeben, um die Lautstärke einzustellen. Im Ansprechen auf einen Empfang der Aufforderung hin, weist das B_Kanal Programm im Block **1013** der **Fig.** 9 die I/F-Platine, die zu dem B-Kanal gehört, der mit dem Benutzer verbunden ist, an,.. die Lautstärke der Information, z. B. der Musik, die gerade über diesen Kanal an den Benutzer übertragen wird, zu erhöhen oder zu verkleinern.

[0055] In bestimmten Anwendungen, z. B. einer Filmreiseberichtanwendung, kann ein Benutzer' den Wunsch haben, einen Liveoperator oder einen Agenten, der zu der Anwendung gehört, zu kontaktieren. Als Folge einer Betrachtung einer Multimediaanwendung, die sich auf einen Filmreisebericht bezieht, könnte der Benutzer z. B. den Wunsch haben, mit einem Reisebüro zu sprechen, um weitere Details über einen Urlaubs-„Platz", über den in der Anwendung berichtet wird, zu ermitteln. Wenn dies der Fall ist, ist es wahrscheinlich, dass die Anwendung dem Benutzer erlauben wird, eine Aufforderung zum Sprechen mit einem Reisebüro, das zu der Anwendung gehört, einzugeben. Demzufolge verzweigt sich das Programm in Ansprechen auf eine derartige Aufforderung hin zum Block **1004** der **Fig.** 9, der dafür ausgelegt ist, um eine Konferenzschaltungsfunktion auszuführen. Eine erweiterte Version des Blocks **1004** ist in **Fig.** 19 dargestellt. Es ist ersichtlich, dass das Unterprogramm oder das Programm, das in **Fig.** 19 dargestellt ist, etwas ähnlich zu dem Unterprogramm ist, das in **Fig.** 10 dargestellt ist. Jedoch arbeitet das Anruf_Überbrücken-Programm
(Bridge_Call_Programme) etwas anders. Das Programm ermittelt (Block **1014-2**) insbesondere, wenn es gestartet wird (Block **1014-1**), einen freien B-Kanal und markiert den Kanal in der Kanalzuordnungstabelle als belegt (busy). Das Programm (Block **1014-3**) sendet dann einen Befehl an die Schnittstellenplatine, um einen Telefonanruf über den B-Kanal zu der Telefonnummer aufzubauen, die in der Aufforderungsnachricht enthalten ist, wobei diese Nachricht in dem vorliegenden Fall von dem Datenserver initiiert wurde. Das Programm umfasst auch in dem Befehl eine Aufforderung für eine Konferenzschaltung (Überbrückung) des neu aufgebauten Anrufs mit dem analogen B-Kanal, der dem Benutzer zugeordnet ist. Das Programm (Block **1014-4**) aktualisiert dann die globale Benutzerinformationstabelle, um den Status der Anwendung zu verfolgen und die Zuordnung des B-Kanals zur Kenntnis zu nehmen. Das Programm kehrt dann zu Block **1001** zurück.

[0056] Es sei an diesem Punkt darauf hingewiesen, dass ab dem Zeitpunkt, zu dem die I/F-Platine die Verbindung zu der dritten Partei (z. B. dem Reisebüro) hergestellt hat, dann der Benutzer und die dritte Partei miteinander kommunizieren können. Wenn die Konversation zwischen dem Benutzer und der dritten Partei abgeschlossen ist, dann kann der Benutzer eine Aufforderung zum Beenden (Aufheben der Überbrückung) der dritten Partei eingeben. Eine derartige Aufforderung wird, wie die meisten Aufforderungen, die von dem Benutzer eingegeben werden, an den B-Kanal Server mit Hilfe der ISDN-Verbindung zwischen dem Terminal des Benutzers und dem Datenserver geliefert. Auf den Empfang einer derartigen Aufforderung hin verzweigt sich das B_Kanalal_Serverprogramm zu der Anruf-Überbrückung-Aufheben-Routine (Unbridge Call Routine), die mit dem Block **1015** dargestellt wird. Eine erweiterte Version des Blocks **1015** ist in **Fig.** 20 darge-

stellt. Insbesondere dann, wenn die Anruf-Überbrückung-Aufheben-Routine gestartet wird (Block **1015-1**) ermittelt sie aus der Informationstabelle des Benutzers die Identität des ISDN B-Kanals, der verwendet wurde, um einen Anruf zu der dritten Partei herzustellen. Das Programm (Block **1015-2**) sendet dann einen Befehl zum Beenden der B-Kanal-Verbindung zu der I/F-Platine, wobei der Befehl den B-Kanal identifiziert. Das Programm (Block **1015-3**) aktualisiert dann die Kanalzuordnungstabelle, um zu zeigen, dass (a) der letztere Kanal frei ist und die Benutzerinformationstabelle zu aktualisieren, dass (b) die Verbindung der dritten Partei getrennt worden ist und dass (c) die Überbrückung des B-Kanals, der dem Benutzer zugewiesen war, aufgehoben wird. Das Programm kehrt dann zum Block **1001** der **Fig.** 9 zurück.

[0057] Wenn der analoge B-Kanal, der zu dem Benutzer gehört, nicht mehr benötigt wird, dann verzweigt sich das Programm zum Block **1016**, FIG. 9. Eine erweiterte Version des Blocks **1016** ist in **Fig.** 21 gezeigt. Wenn das Programm gestartet wird (Block **1016-1**, FIG. 21), sendet das Programm einen Trennungsbefehl (Block **1016-2**) an die I/F-Platine zum Trennen des zugehörigen B-Kanals, wobei der Befehl den letzteren Kanal identifiziert. Die I/F-Platine verwendet wiederum die Information in dem Trennungsbefehl, um eine Signalisierungsnachricht zu bilden, und sendet die Signalisierungsnachricht über den zugehörigen Signalisierungs- (D) – Kanal. Das Programm (Block **1016-3**) aktualisiert dann die Zuordnungstabelle, um zu zeigen, dass der Kanal frei ist und aktualisiert die Benutzerinformationstabelle entsprechend. Das Programm kehrt dann zum Block **1001** zurück.

[0058] Das Programm, das den Datenserver ansteuert ist etwas ähnlich zu dem Programm, das den Audioserver ansteuert, wie nachstehend ersichtlich ist. Insbesondere dann, wenn Energie an den Datenserver angelegt wird und er „heraufgefahren" wird, wird ein einfaches Programm (nicht gezeigt) gestartet, um den in **Fig.** 22 gezeigten Prozess zu initiieren und den Prozess mit einem jeweiligen der B-Kanäle, die die Multimediaplattform (**Fig.** 1) bedienen, in Zusammenhang zu bringen. Insbesondere das Programm der **Fig.** 22 ist eine einfache Schleifenbildungsanordnung, die zunächst in eine B-Kanalverwaltungsroutine (Block **2001**) übergeht. Wenn die letztere Routine ihre Task beendet dann geht das Programm in eine B-Kanalservenoutine (Block **2002**) über. Eine erweiterte Version des Blocks **2001** ist in **Fig.** 23 gezeigt. Wenn das Programm gestartet wird (Block **2003**), dann wartet das Programm insbesondere auf dem Empfang einer Nachricht von dem zugehörigen B-Kanal. Auf einen Empfang einer Nachricht von der I/F-Platine hin, die mit dem zugehörigen B-Kanal verbunden ist, verzweigt sich das Programm an eine Anzahl von unterschiedlichen Routinen auf Grundlage des Typs der Nachricht, den es empfangen hat. Wenn ein Nachrichtentyp unbekannt ist, dann verzweigt sich das Programm zum Block **2004**, wo es den Fehler an das OS-System **50** übergibt, das die empfangene Nachricht zur Verarbeitung von einem Plattformadministrator speichert. Das Programm kehrt dann zum Block **2003** zurück, um auf einen Empfang einer neuen Nachricht über seinen zugehörigen B-Kanal zu warten.

[0059] Wenn sich die empfangene Nachricht auf einen ankommenden Anruf bezieht, dann verzweigt sich das Programm zu einer Verbindungsroutine (Block **2005**). Eine erweiterte Version des Blocks **2005** ist in **Fig.** 24 gezeigt. Wenn sie gestartet ist (Block **2005-1**), dann sendet die Verbindungsroutine eine Nachricht an die I/F-Platine, die mit dem zugehörigen B-Kanal gekoppelt ist, um eine Antwortüberwachungs-Signalisierungsnachricht über den D-Kanal als Vorgehensweise zum Herstellen einer Verbindung zwischen dem B-Kanal und der Schnittstellenplatine zu senden. Die Routine prüft dann nach (Block **2005-3**), um nachzusehen, ob die anrufende Partei (z. B. das Terminal **T1**, **Fig.** 1) bereits mit einem anderen B-Kanal verbunden ist (d. h., die empfangene Nachricht eine Aufforderung nach einem zweiten B-Kanal ist). Wenn dies nicht der Fall ist, dann aktiviert bzw. erzeugt die Routine (das Programm) (Block **2005-4**) einen B_Kanal Server, um die neue Verbindung zu bedienen und aktualisiert dann die B_Kanal Zuordnungstabelle (Block **2005-5**), um zu zeigen, dass der Kanal belegt ist, wie voranstehend diskutiert. Die Routine geht dann zurück zum Block **2003** der **Fig.** 23. Wenn die anrufende Partei (Gegenstelle) bereits verbunden ist, dann fügt die Routine (Block **2005-6**) eine sogenannte Kanalnachricht zu der Warteschlange des B-Kanal Servers, der zu dem B-Kanal gehört, von dem die Nachricht im Block **2003** empfangen wurde, hinzu.

[0060] Wenn eine ankommende Nachricht eine Trennungsaufforderung ist, dann geht das Programm am Block **2003** der **Fig.** 23 anstelle davon zum Block **2006**, wo es die B-Kanal Zuordnungstabelle aktualisiert, um zu zeigen, dass der B-Kanal frei ist. Wenn jedoch die ankommende Nachricht darauf gerichtet ist, einen Anruf durchzuführen, dann geht der B_Kanal Manager weiter zum Block **2007**. Eine erweiterte Version des Blocks **2007** ist in **Fig.** 31 gezeigt. Wenn das Programm gestartet ist, geht es vom Block **2007-1** zum Block **2007-2**, wo es einen freien B-Kanal dem Anruf zuordnet, der in der empfangenen Nachricht identifiziert wird. Das Programm (der Block **2007-3**) weist dann die I/F-Platine an, die Telefonnummer zu wählen, die in der Nachricht enthalten ist und verarbeitet dann (Block **2007-4**) Statusinformation, die sich auf die Durchführung des Anrufs bezieht, die von der I/F-Platine zurückgegeben wird. An diesem Punkt kehrt das Programm zum Block **2003** zurück.

[0061] Es sei darauf hingewiesen, dass dann, wenn ein in **Fig.** 25 gezeigtes B Kanal Serverprogramm im Block **2005-4** gestartet bzw. aktiviert wird, dann sämtliche Nachrichten, die danach über den zugeordne-

ten B-Kanal empfangen werden, an den erzeugten bzw. aktivierten B_Kanal Server übergeben werden und nicht an das B_Kanal Verwalterprogramm (Managerprogramm). Wenn es so hervorgebracht bzw. aktiviert worden ist (Block **2005-4**, FIG. 24), wird das Programm insbesondere, **Fig.** 25, in Block **2400** gestartet, wo es auf den Empfang einer Nachricht über den zugeordneten B-Kanal wartet. Auf den Empfang einer Nachricht hin verzweigt sich das Programm an eine Anzahl von unterschiedlichen Routinen (Blöcke **2401** bis **2404**) auf Grundlage des Typs der Nachricht, die empfangen wurde. Wenn die Nachricht nicht eine der bekannten Typen ist, dann markiert das Programm (Block **2404**) das Ereignis als einen Fehler und übergibt den Fehler an das OS-System **50**. Das Programm kehrt dann zum Block **2400** zurück, um auf den Empfang der nächsten Nachricht zu warten. Wenn sich der Nachrichtentyp auf das Anwendungscontrollerprogramm (nachstehend diskutiert) bezieht, dann speichert das Programm (**2401**) die Nachricht in der Warteschlange dieses Programms. Wenn sich die Nachricht auf eine Funktion bezieht, die zu dem Audioserverprogramm gehört, dann speichert das Programm (Block **2402**) in einer ähnlichen Weise die Nachricht in der Warteschlange dieses Programms. Wenn sich die Nachricht auf eine Datenserverfunktion bezieht, dann speichert das Programm (Block **2403**) die Nachricht in der Warteschlange eines sogenannten Datenserver-Befehlsprozessors. **Fig.** 25 zeigt, dass jeder der Blöcke **2401** bis **2404** zu dem Block **2400** zurückkehrt, nachdem der Block seine jeweilige Task beendet hat.

[0062] Eine erweiterte Version des Daten_Server_Befehlsprogramms (data_server_command Programms oder data_server_cmd) (Block **2403**) ist in einer Flussdiagrammform in **Fig.** 26 gezeigt. Wenn das Programm in Block **2500** so gestartet wird, initialisiert das Programm insbesondere eine Anzahl von lokalen Variablen. Das Programm überprüft dann seine zugehörige Warteschlange, um nachzusehen, ob eine Nachricht darin gespeichert ist. Wenn dies nicht der Fall ist, endet das Programm und wird danach nach Ablauf einer vorgegebenen Zeitspanne, z. B. 25 Millisekunden, erneut angefangen. Wenn eine Nachricht in der Warteschlange ist, dann verzweigt sich das Programm, genau so wie der voranstehend diskutierte B_Kanal Prozess, in Abhängigkeit von dem Nachrichtentyp an eine einer Anzahl von unterschiedlichen Routinen, die jeweils mit den Blöcken **2501** bis **2514** dargestellt sind. Wenn zunächst der Block **2514** und dann jeder der Blöcke **2501** bis **2513** betrachtet wird, verzweigt sich das Programm zum Block **2514** für den Fall, dass der Nachrichtentyp nicht bekannt ist. Im Block **2514** übergibt das Programm den Fehler an das OS-System **50** und kehrt dann zum Block **2400** zurück.

[0063] Wenn die Nachricht anzeigt, dass ein zweiter B-Kanal dem Benutzer zum Übertragen von Daten zugeordnet worden ist, dann verzweigt sich das Programm zum Block **2501**, wo es die zugehörige Informationstabelle des Benutzers aktualisiert, um diese Transaktion zu verfolgen, und kehrt dann zum Block **2400** zurück. Wenn die Nachricht sich auf eine Aufforderung für eine Verbindung bezieht, dann verzweigt sich das Programm zu Block **2502**, um die Verbindung herzustellen. Insbesondere überprüft das Programm zunächst die Nachricht, um zu bestimmen, ob sie eine Sprachverbindung anfordert und übergibt die Nachricht an den analogen Server, wenn dies der Fall ist. Wenn dies nicht der Fall ist, dann überprüft das Programm die voranstehend erwähnte globale Informationstabelle, um festzustellen, ob zwei B-Kanäle dem Benutzer bereits zugewiesen worden sind. Wenn dies der Fall ist, dann prüft das Programm nach, um nachzusehen, ob beide Kanäle als Datenkanäle zugeordnet worden sind. Wenn dies der Fall ist, dann kann das Programm nicht auf die Verbindungsnachricht ansprechen und kehrt zum Block **2400** zurück. Ansonsten führt das Programm eine Zuweisung (Zuordnung) eines freien B-Kanals an den Benutzer in der voranstehend beschriebenen Weise aus oder ändert den Status eines B-Kanals, der bereits dem Benutzer zugewiesen ist, und aktualisiert dann die globalen Informationskanalzuordnungstabellen, je nach Fall, um zu zeigen, dass der zugewiesene Kanal als ein Datenkanal in Verwendung ist. Das Programm kehrt dann zum Block **2400** zurück.

[0064] Wenn die Anwendung, die der Benutzer auf dem Terminal **T1** ablaufen lässt, nicht mehr einen B-Kanal erfordert, der gerade entweder in einem Audio- oder Datenmodus verwendet wird, dann wird die Nachricht eine Aufforderung zum Trennen des Kanals sein, wodurch bewirkt wird, dass das Programm weiter zum Block **2503** geht. Im Block **2503** prüft das Programm, um nachzusehen, ob die Nachricht eine Aufforderung zum Trennen eines B-Kanals ist, der für Sprache verwendet worden ist. Wenn dies der Fall ist, dann übergibt das Programme die Nachricht an den Audioserver und kehrt dann zum Block **2400** zurück Ansonsten trennt das Programm den Kanal, indem es bewirkt, dass er in einen freien Modus (idle mode) gebracht wird, wie voranstehend im Zusammenhang mit dem Audioserver diskutiert. Das Programm aktualisiert die verschiedenen voranstehend erwähnten Tabellen und kehrt dann zum Block **2400** zurück.

[0065] Wenn die Nachricht eine Aufforderung von einem Anwendungsprogramm zum Herunterladen einer bestimmten Datendatei an das Terminal des Benutzers ist, dann geht das Programm weiter zum Block **2504**. Im Block **2504** bestimmt das Programm ob ein „Herunterladeprozess" gegenwärtig im Zusammenhang mit dem Ablaufenlassen der von dem Benutzer gewählten Anwendung ist. Wenn dem so ist, dann fügt das Programm die Aufforderung zu der Warteschlange dieses Herunterladeprozesses hinzu und kehrt dann zum Block **2400** zurück. Ansonsten erzeugt das Programm einen Herunterladeprozess,

um die Datendatei, die in der empfangenen Nachricht identifiziert wird, herunter zu laden. Das Programm speichert dann die empfangene Nachricht in der Warteschlange des neu erzeugten Herunterladeprozesses und kehrt zu dem Block **2400** zurück. Ein Flussdiagramm des Herunterladeprozesses ist in **Fig.** 27 gezeigt. Wenn es begonnen wird (Block **2600**), dann initialisiert das Programm insbesondere verschiedene Variablen, die es verwendet und bestimmt dann (Block **2601**), ob seine Aufforderungswarteschlange leer ist. Wenn dem so ist, dann endet das Programm. Ansonsten baut das Programm (Block **2602**) eine Liste (INPROG_LIST) der Namen der Dateien, die es an das Benutzerterminal herunterladen wird. Mit dieser Vorgehensweise betrachtet das Programm die gegenwärtige Konfiguration der zwei B-Kanä1e (oder eines B-Kanals) als zwischen die Multimediaplatlform und das Benutzerterminal geschaltet, wie nachstehend diskutiert. Das Programm (Block **2603**) prüft dann nach, um nachzusehen, ob seine INPROG_LIST leer ist. Wenn die Liste nicht leer ist, dann geht das Programm in ein Unterprogramm XMIT_LIST über, um jede Datei zu übertragen, die in der INPROG_List identifiziert wird. Wenn das Programm feststellt, das die INPROG_LIST leer ist, dann wartet es in einer ähnlichen Weise (Block **2605**) auf eine Benachrichtigung von dem B-Kanal Server, dass eine neue Aufforderung in der Warteschlange gespeichert worden ist und kehrt zum Block **2601** auf den Empfang einer derartigen Benachrichtigung hin zurück.

[0066] Das XMIT_LIST Unterprogramm oder die Routine ist in **Fig.** 28 gezeigt. Wenn es angefangen wird (Block **2604-1**), dann entfernt das Programm insbesondere einen Dateinnamen von der INPROG_LIST und beginnt das Entladen der Daten, die die Datei bilden, von dem Diskspeicher. Durch eine derartige Vorgehensweise speichert das Programm die entladenen Daten in zugehörigen Sendepuffern (xmit_buffer), die zu der I/F-Platine gehören, die dem bestimmten B-Kanal (die bestimmte B-Kanäle) bedient. Das heißt, wenn gerade zwei B-Kanäle verwendet werden, dann werden die Daten in den Sendepuffern der I/F-Platine, die für diese zwei Kanäle verwendet werden, gespeichert. In dieser Weise wird die Übertragungsbandbreite verdoppelt, wodurch die Übertragung der Datendatei beschleunigt wird. Wenn nur ein B-Kanal gerade für Daten verwendet wird, dann wird die entladene Datei in dem Sendepuffer der UF-Platine, der für diesen Kanal verwendet wird, gespeichert. Wenn ferner eine Anzahl von Dateien herunter geladen werden soll, dann wird die Bandbreite der zwei Kanäle oder eines Kanals dem Herunterladen von diesen Dateien zugeordnet. Wenn zum Beispiel zwei Dateien herunter geladen werden sollen und zwei B-Kanäle gerade verwendet werden, dann kann die eine Datei einem Kanal zugeordnet werden und die andere Datei kann dem anderen Kanal zugeordnet werden. Als ein anderes Beispiel kann eine Datei eineinhalb Kanälen zugeordnet werden. Was dies bedeutet ist, dass ein Teil der Datei über einen Kanal übertragen wird und der andere Teil der Datei und eine andere Datei über den anderen Kanal multiplexiert werden. Wenn als ein anderes Beispiel nur ein B-Kanal verfügbar ist, dann kann ein Dreiviertel der Bandbreite des Kanals speziell für eine Datei verwendet werden und die verbleibende Bandbreite kann für eine andere Datei speziell verwendet werden. Eine derartige gemeinsame Verwendung der B-Kanal Bandbreite wird von dem Programm berechnet und in der INPROG_LIST niedergelegt.

[0067] Wenn während einer derartigen Entladung bei einer Fortsetzung das Programm (Block **2604-3**) Daten antrifft, die das Dateieende (End OF the File, EOF) anzeigen, dann schließt das Programm (Block **2604-7**) die Datei und löscht die Identität der Datei von der INPROG_LIST und der Aufforderungswarteschlange. Das Programm (Block **2604-4**) sendet dann einen Befehl an die I/F-Platine, wobei sie die Platine anweist, die Inhalte des xmit_uffers (xmit_Buffer) zu übertragen. Das Programm (Block **2604-5**) prüft dann, um nachzusehen, ob es eine Benachrichtigungsnachricht empfangen hat, die anzeigt, dass eine neue Anforderung in seiner zugehörigen Aufforderungswarteschlange gespeichert worden ist. Wenn dem so ist, dann kehrt das Programm zum Block **2601** der **Fig.** 27 zurück. Ansonsten prüft das Programm (Block **2604-6**), um nachzusehen, ob die INPROG_LIST leer ist und kehrt dann zum Block **2601** zurück, wenn dies der Fall ist. Ansonsten kehrt das Programm zum Block **2604-2** zurück Wenn das Programm (**Fig.** 26) zum Block **2505**, **2506** oder **2507** weiter geht, dann weist es jeweils den Herunterladeprozess (**Fig.** 27) an, das Herunterladen einer bestimmten Datei zu löschen, zu pausieren oder wieder aufzunehmen. Wenn die empfangene Nachricht sich auf das Heraufladen einer Datendatei von dem Benutzerterminal an die Multimediaplattform bezieht, dann geht das Programm zum Block **2508**, wo es wartet, um eine Datendatei von dem Benutzerterminal für eine Speicherung in der Multimediaplattform, d. h. dem Diskspeicher, zu empfangen. Das heißt, das Programm geht in einen Heraufladeprozess über und speichert die empfangene Nachricht in der Warteschlange des Heraufladeprozesses.

[0068] Der Prozess, der im Block **2508** begonnen wird, ist in **Fig.** 29 gezeigt. Es lasst sich ersehen, dass der Heraufladeprozess in etwa ähnlich zu dem Herunterladeprozess (**Fig.** 27) ist. Deshalb wird **Fig.** 29 kurz erläutert. Wenn das Programm begonnen wird (Block **2800**) initialisiert es in ähnlicher Weise verschiedene Variablen, die es verwendet, und endet dann, wenn seine zugehörige Aufforderungswarteschlange leer ist. Ansonsten konstruiert das Programm (Block **2802**) eine Liste (INPROG_LIST) der Nachrichten (Namen der Dateien, die von dem Benutzer empfangen werden), die in seiner zugehörigen Warteschlange enthalten sind und auf Grundlage der Konfiguration der zugeordneten B-Kanäle.

Das Programm (Block **2803**) prüft dann seine INPROG_LIST und wartet (Block **2805**) auf den Empfang einer Benachrichtigungsnachricht, wenn die Liste leer ist. Ansonsten geht das Programm in einen Prozess (RCV_LIST) für den Empfang einer Datei über, die von dem Benutzerterminal übertragen werden wird.

[0069] Der RCV_LIST Prozess ist in Flussdiagrammform in **Fig.** 30 dargestellt. Es sei darauf hingewiesen, dass die I/F-Platine die Einrichtung ist, die eine Datei von dem zugehörigen B-Kanal empfängt und die Datei in einem zugehörigen Empfangspuffer (RCV_BUF) speichert, wenn die Datei empfangen wird. Wenn der Puffer voll ist, dann benachrichtigt die I/F-Platine den RCV_LIST Prozess. Der RCV_LIST Prozess oder das Programm (Block **2901**) leert im Ansprechen darauf den Puffer. Das Programm (Block **2902**) speichert dann die Daten, die von dem Pufferspeicher entladen werden, in dem Diskspeicher **61** (in der Disk bzw. Platte). Es sei darauf hingewiesen, dass derartige Daten mit einer oder mehreren Dateien im Zusammenhang stehen können, wobei die Kanalkonfiguration anzeigt, wie die Kanalbandbreite unter den Dateien gemeinsam verwendet wurde, wenn mehr als eine Datei gerade herauf geladen wird. Wenn während eines derartigen Herauflade- und Speichervorgangs das Programm einen EOF (Block **2903**) antrifft, dann schließt es die Datei (Block **2906**) und entfernt den Dateinamen dann von seiner zugehörigen Warteschlange und der INPROG_LIST. Das Programm kehrt dann zum Block **2801** der **Fig.** 29 zurück. Ansonsten prüft das Programm (Block **2904**) um nachzusehen, ob es darüber informiert worden ist, dass eine neue Heraufladeaufforderungsnachricht in seiner Warteschlange gespeichert worden ist und kehrt nach 2801 zurück, wie gezeigt, wenn dies der Fall ist. Wenn dies nicht der Fall ist, dann kehrt das Programm (Block **2905**) zum Block **2901** zurück, wenn es feststellt, dass die INPROG_LIST (INPROG_LISTE) nicht leer ist. Ansonsten kehrt es zum Block **2801** zurück.

[0070] Wenn das Programm bei **Fig.** 26 zum Block **2509**, **2510** oder **2511** fortschreitet, dann weist es jeweils den Heraufladeprozess (**Fig.** 29), das Heraufladen einer bestimmten Datei zu pausieren, wieder aufzunehmen oder zu löschen.

[0071] Der Block **2512**. der **Fig.** 26 stellt eine Schnittstelle zwischen dem Datenserverprogramm und einem Anwendungsprogramm bereit. Das heißt, der Block **2512** ist die Einrichtung, mit der das Programm, als Reaktion auf eine Aufforderung, die von dem Anwendungsprogramm empfangen wird, das letztere immer dann, wenn eine bestimmte Herunterladung oder Heraufladung abgeschlossen worden ist, benachrichtigt. Insbesondere durchsucht das Programm im Block **2512** die Aufforderungs-Warteschlangen, die zu den Herunterladungs- und Heraufladungsprozessen gehören, nach dem Namen einer Datei, die in der Aufforderung enthalten ist, die von dem Anwendungsprogramm empfangen wird. Wenn das Programm den Namen der betreffenden Datei in irgendeiner Warteschlange findet, dann gibt es ein erstes Flag (eine Marke), d. h. einen Wert von unwahr, an das Anwendungsprogramm zurück, um anzuzeigen, dass der, Datentransfer nicht abgeschlossen worden ist (oder der Transfer gestoppt worden ist). Ansonsten gibt das Programm ein zweites Flag, z. B. einen Wert von wahr, an das Anwendungsprogramm zurück.

[0072] Das Datenserverprogramm der **Fig.** 26 geht zum Block **2513**, wenn der Typ von Nachricht eine Aufforderung zum Neukonfigurieren von zwei B Kanälen ist, die dem Benutzer zugeordnet worden sind. Das heißt, die Aufforderung ist typischerweise von einem Anwendungsprogramm zum Neukonfigurieren der zwei B_Kanäle, z. B. um einen der B_Kanäle fallen zu lassen. Insbesondere aktualisiert das Programm in Block **2513** die Benutzerinformationstabelle und die Kanalzuordnungstabelle, um der Neukonfigurationsaufforderung angepasst zu sein. Das Programm benachrichtigt dann die Herunterladungsund Heraufladungsprozesse darüber, dass eine Kanalneukonfiguration aufgetreten ist, und hält den Ablauf der Herunterladungs- und Heraufladungsprozesse während einer derartigen Aktualisierung an.

[0073] Der Übergang in eine Anwendung im Ansprechen auf eine Benutzerauswahl ist unter der Steuerung eines Anwendungscontrollerprogramms, welches auf dem Anwendungscontroller vorhanden ist. Wenn der Anwendungscontroller hochgefahren wird, dann wird insbesondere das Anwendungscontrollerprogramm (APP_CTLR) **Fig.** 32 im Block **3000** gestartet, wo es auf einen Empfang einer Aufforderungsnachricht wartet, die eine Benutzerauswahl einer bestimmten Anwendung oder das Stoppen, Pausieren oder Wiederaufnehmen einer bereits aktivierten Anwendung identifiziert. Auf einen Empfang einer Aufforderung hin verzweigt sich das Programm (Block **3001**) insbesondere an eines einer Anzahl von unterschiedlichen Unterprogrammen (Blöcke **3002** bis **3006**) auf Grundlage des Typs der Nachricht, die empfangen wird. Wenn der Nachrichtentyp nicht bekannt ist, dann nimmt das Programm (Block **3006**) an, dass die Nachricht ein Fehler ist und übergibt sie an das OS-System **50** für eine Verarbeitung. Das Programm kehrt zum Block **3000** zurück, um auf einen Empfang einer nächsten Anwendungs-bezogenen Aufforderung zu warten.

[0074] Wenn die Aufforderung zum Starten einer bestimmten Anwendung vorgesehen ist, dann geht das Programm zum Block **3002**, wo es die Anwendung aktiviert. Zusätzlich übergibt das Programm die Konfiguration der B-Kanäle, die zu dem Benutzer gehören, an das aktivierte Anwendungsprogramm. Das Controllerprogramm aktualisiert dann die Benutzerinformationstabelle, um die gewählte Anwendung zu identifizieren. Das Programm kehrt dann zum Block **3000** zurück. Wenn die Aufforderung auf das Stoppen eines aktivierten Anwendungsprogramms gerichtet ist, dann sendet der Controller (Block **3003**)

eine Nachricht diesbezüglich an das Anwendungsprogramm und aktualisiert die Benutzerinformationstabelle entsprechend. In ähnlicher Weise sendet (Block **3004** oder **3005**) das Controllerprogramm im Ansprechen auf eine Aufforderung zum Pausieren (oder zum Wiederaufnehmen) eine Nachricht in dieser Hinsicht an die Anwendung. Die Anwendung spricht wiederum auf den Empfang der Aufforderung (über den Block **3003**, **3004** oder **3005** gesendet) durch Ausführen der angeforderten Aktion an.

ANHANG A

[0075] Die folgenden Zeilen weisen den Audioserver an (fordert diesen auf), eine Sprachdatei abzuspielen.
[0076] int_plag_audio_file(int handle, char *ap, char *tune)
[0077] Die Variable handle ist der Wert, der von der Verbindungsfunktion (connect function) zurückgegeben wird, wenn die Verbindung aufgebaut wurde, wie ferner nachstehend beschrieben wird; ap ist ein Zeiger auf eine Zeichenkette, die den Anwendungsnamen spezifiziert; tune ist ein Zeiger auf eine Zeichenkette, die den Audiodateinamen spezifiziert über eine vorher aufgebaute B-Kanalsprachverbindung.
[0078] Die folgende Zeile weist den Audioserver an, die Wiedergabe einer Sprachdatei an ihrem gegenwärtigen Punkt zu unterbrechen (zu pausieren).
[0079] int pause_audio_playback(int handle)
[0080] Die folgende Zeile weist den Audioserver an, die Wiedergabe einer pausierten Sprachdatei wieder aufzunehmen.
[0081] int resume_audio_playback(int handle)
[0082] Die folgende Zeile weist den Audioserver an, an den Start der nächsten Sprachdatei in der Wiedergabeliste zu springen und diese wiederzugeben.
[0083] int skip_audio_playback (int handle)
[0084] Die folgende Zeile weist den Audioserver an, die Lautstärke des Audio, der über eine B-Kanal-Sprachverbindung übertragen wird, einzustellen. Die Lautstärke kann erhöht oder verringert werden.
[0085] int adjust_volume(int handle)
[0086] Die folgende Zeile weist den Audioserver an, die Wiedergabe einer Sprachdatei, die gerade über eine B-Kanal-Sprachverbindung wiedergegeben wird, zu unterbrechen.
[0087] int stop_audio_playback(int handle)
[0088] Die folgende Zeile weist den Audioserver an, eine andere Sprachverbindung auf eine B-Kanal-Sprachverbindung zu überbrücken.
[0089] int bridge_call(int handle)
[0090] Die folgende Zeile weist den Audioserver an, eine Sprachverbindung zu entfernen, die vorher auf den Anruf überbrückt worden ist.
[0091] int unbridge_call(int handle)
[0092] Die folgende Zeile weist den Audioserver an, Audio, das gerade über eine Sprachverbindung empfangen wird, in eine Sprachdatei aufzuzeichnen.
[0093] record_audio(int handle)
[0094] Die folgende Zeile weist den Audioserver an, die Aufzeichnung einer Sprachdatei an ihrem gegenwärtigen Punkt zu pausieren.
[0095] int pause_audio_record(int handle)
[0096] Die folgende Zeile weist den Audioserver an, die Aufzeichnung einer pausierten Sprachdatei wiederaufzunehmen.
[0097] int resume_audio_record(int handle)
[0098] Die folgende Zeile weist den Audioserver an, das Aufzeichnen einer Sprachdatei zu stoppen.
[0099] int stop_audio_record(int handle)
[0100] Für die obigen Befehle ist „handle" der Wert, der von der Verbindungsfunktion (connect function) zurückgegeben wird, wenn die Verbindung hergestellt wurde, und gibt 0 zurück, wenn erfolgreich, ansonsten eine -1.

Patentansprüche

1. Multimediasystem (**100**), umfassend:
eine Einrichtung zum Koppeln des Systems mit einem Telekommunikationsnetz (**200**) und
einer Einrichtung zum Speichern von unterschiedlichen Multimedianwendungen in einem Speicher (**60**, **61**) und Darbieten einer einzelnen der Anwendungen an einem Teilnehmer (**T1**) über das Telekommunikationsnetz (**200**) im Ansprechen auf den Empfang einer Aufforderung, die die besagte eine Anwendung identifiziert, über das Telekommunikationsnetz (**200**);
wobei das Multimediasystem dadurch gekennzeichnet ist, dass es eine Einrichtung umfasst, ansprechend auf einen Empfang eines Anrufs von einem Anbieter einer Multimediaanwendung, um dann die Multimediaanwendung über das Telekommunikationsnetz zu empfangen und um die empfangene Multimediaanwendung in dem Speicher so zu speichern, dass der Teilnehmer die letztere Multimediaanwendung anfordern kann;
und dass es ferner eine Anwendungsentwicklungsplattform (**150**) umfasst, die arbeitet, um auf einen Telefonanruf von dem Anbieter über das Netz (**200**) anzusprechen und die arbeitet, um über das Netz (**200**) die gespeicherte Multimediaanwendung an dem Anbieter zu präsentieren, um dem Anbieter zu ermöglichen, irgendwelche Fehler zu korrigieren, die in der Anwendung enthalten sein können, und um dann von dem Anbieter für eine Speicherung in einem Speicher die korrigierte Multimediaanwendung zu empfangen.

2. System nach Anspruch 1, wobei die Anwendungsentwicklungsplattform (**150**) ferner arbeitet, um auf einen Telefonanruf von dem Anbieter über das Netz (**200**) anzusprechen und arbeitet, um dem Anbieter zu ermöglichen, auf die gespeicherte Multimediaanwendung zuzugreifen, sogar wenn die gespeicherte Multimediaanwendung nicht vollständig entwickelt und/oder debugged ist.

3. System nach Anspruch 2, wobei die Anwendungsentwicklungsplattform (**150**) ferner eine Ein-

richtung zum Empfangen eines Segments der Multimediaanwendung, die vorher einen oder mehrere Fehler enthalten hat und die durch den Anbieter als Folge des Zugriffs korrigiert worden sind, über das Netz von dem Anbieter umfasst.

4. System nach Anspruch 1, wobei die Anwendungsentwicklungsplattform (**150**) ferner arbeitet, um auf einen Telefonanruf von dem Anbieter über das Netz anzusprechen und arbeitet, um mit dem Anbieter über das Netz gekoppelt zu werden, so dass der Anbieter auf die Multimediaanwendung zugreifen kann, als ob der Anbieter der Teilnehmer (**T1**) wäre, aber für den Zweck einer Korrektes von irgendwelchen Fehlern, die in der Anwendung enthalten sein können; und um dann von einem Anbieter für einen Speicherungsspeicher die korrigierte Multimediaanwendung zu empfangen.

Es folgen 23 Blatt Zeichnungen

Anhängende Zeichnungen

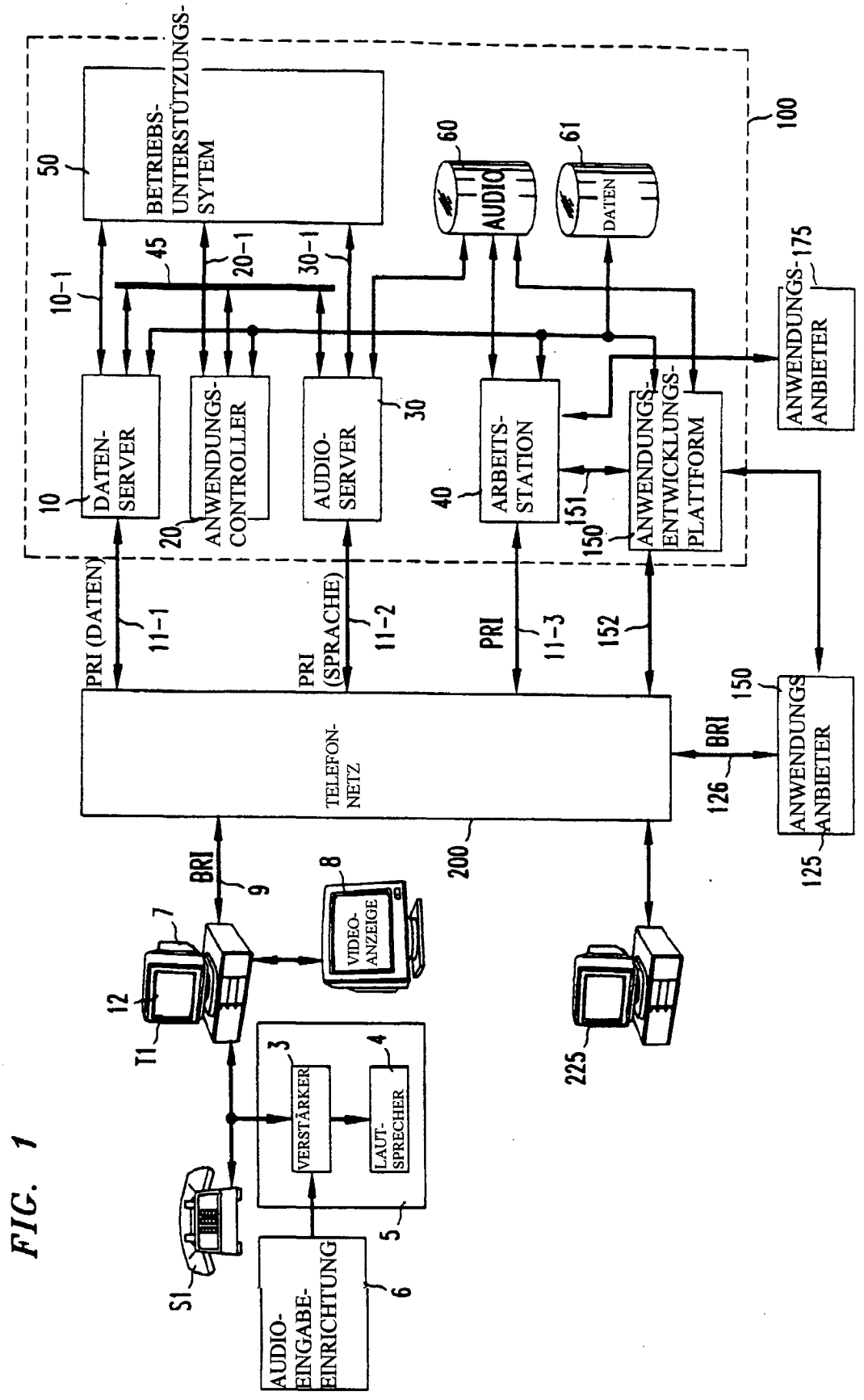


FIG. 1

FIG. 2

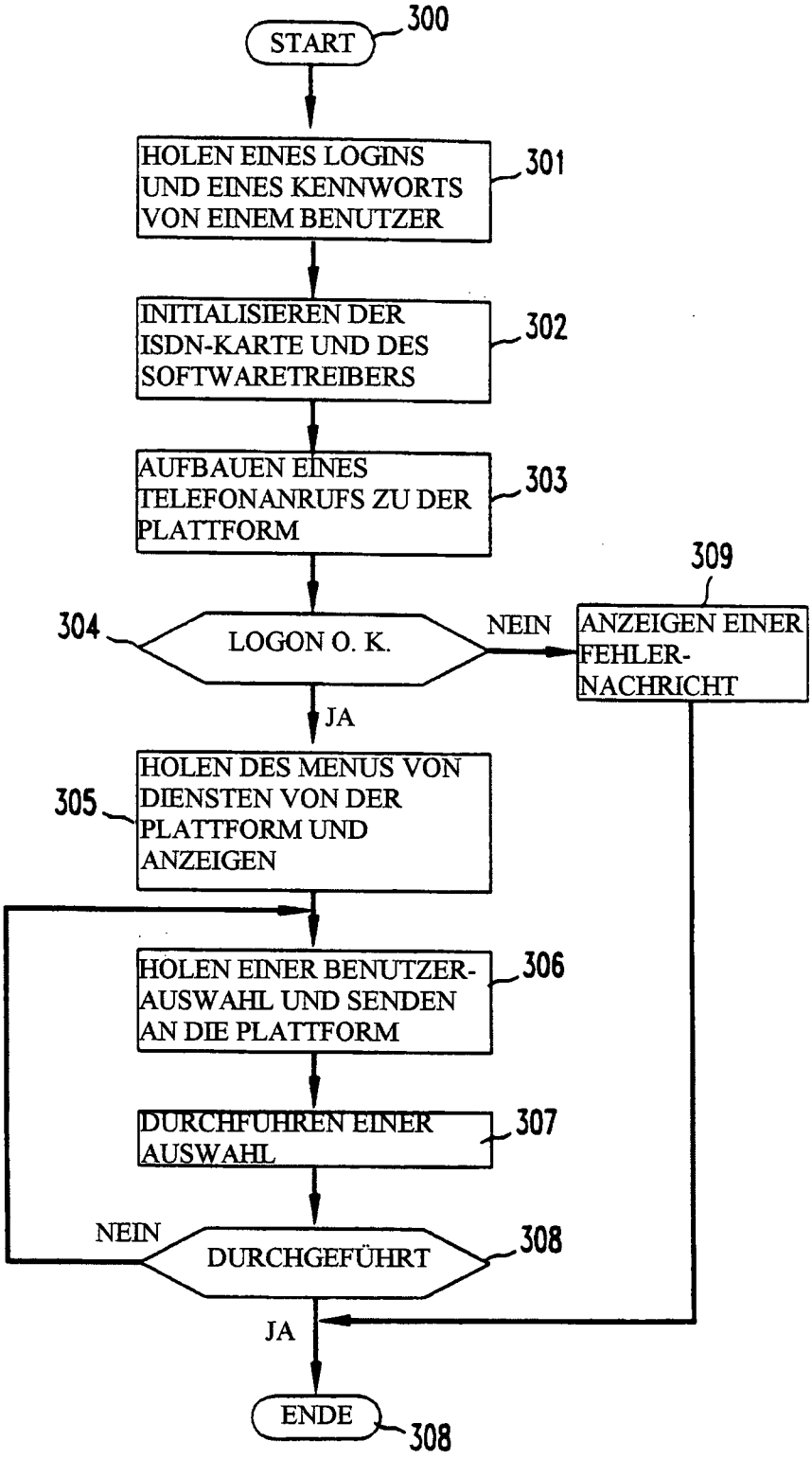

START
300
HOLEN EINES LOGINS UND EINES KENNWORTS VON EINEM BENUTZER
301
INITIALISIEREN DER ISDN-KARTE UND DES SOFTWARETREIBERS
302
AUFBAUEN EINES TELEFONANRUFS ZU DER PLATTFORM
303
309
304
LOGON O. K.
NEIN
ANZEIGEN EINER FEHLER-NACHRICHT
JA
HOLEN DES MENUS VON DIENSTEN VON DER PLATTFORM UND ANZEIGEN
305
HOLEN EINER BENUTZER-AUSWAHL UND SENDEN AN DIE PLATTFORM
306
DURCHFUHREN EINER AUSWAHL
307
NEIN
DURCHGEFÜHRT
308
JA
ENDE
308

FIG. 3

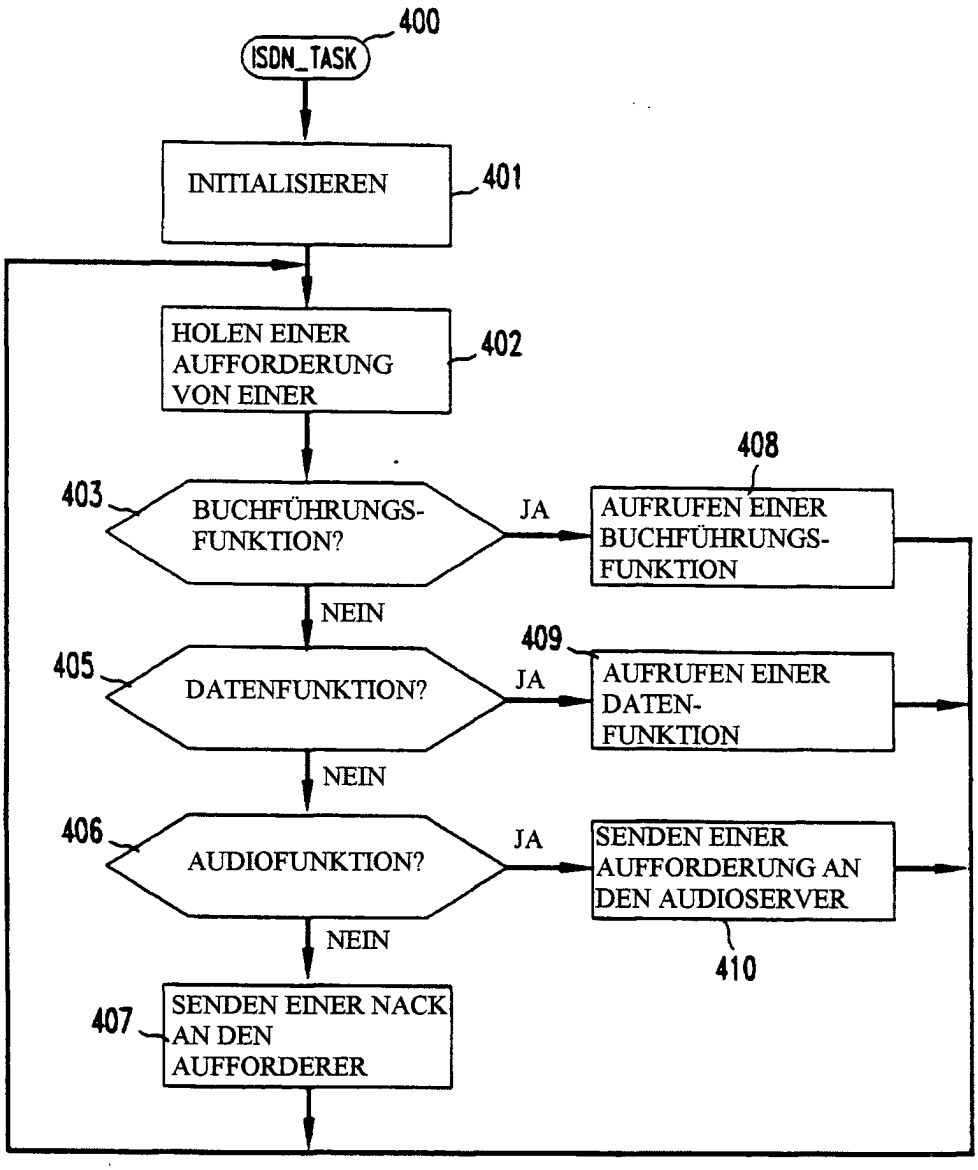

400
ISDN_TASK
INITIALISIEREN
401
HOLEN EINER AUFFORDERUNG VON EINER
402
403
BUCHFÜHRUNGS-FUNKTION?
JA
408
AUFRUFEN EINER BUCHFÜHRUNGS-FUNKTION
NEIN
405
DATENFUNKTION?
JA
409
AUFRUFEN EINER DATEN-FUNKTION
NEIN
406
AUDIOFUNKTION?
JA
SENDEN EINER AUFFORDERUNG AN DEN AUDIOSERVER
410
NEIN
407
SENDEN EINER NACK AN DEN AUFFORDERER

FIG. 4

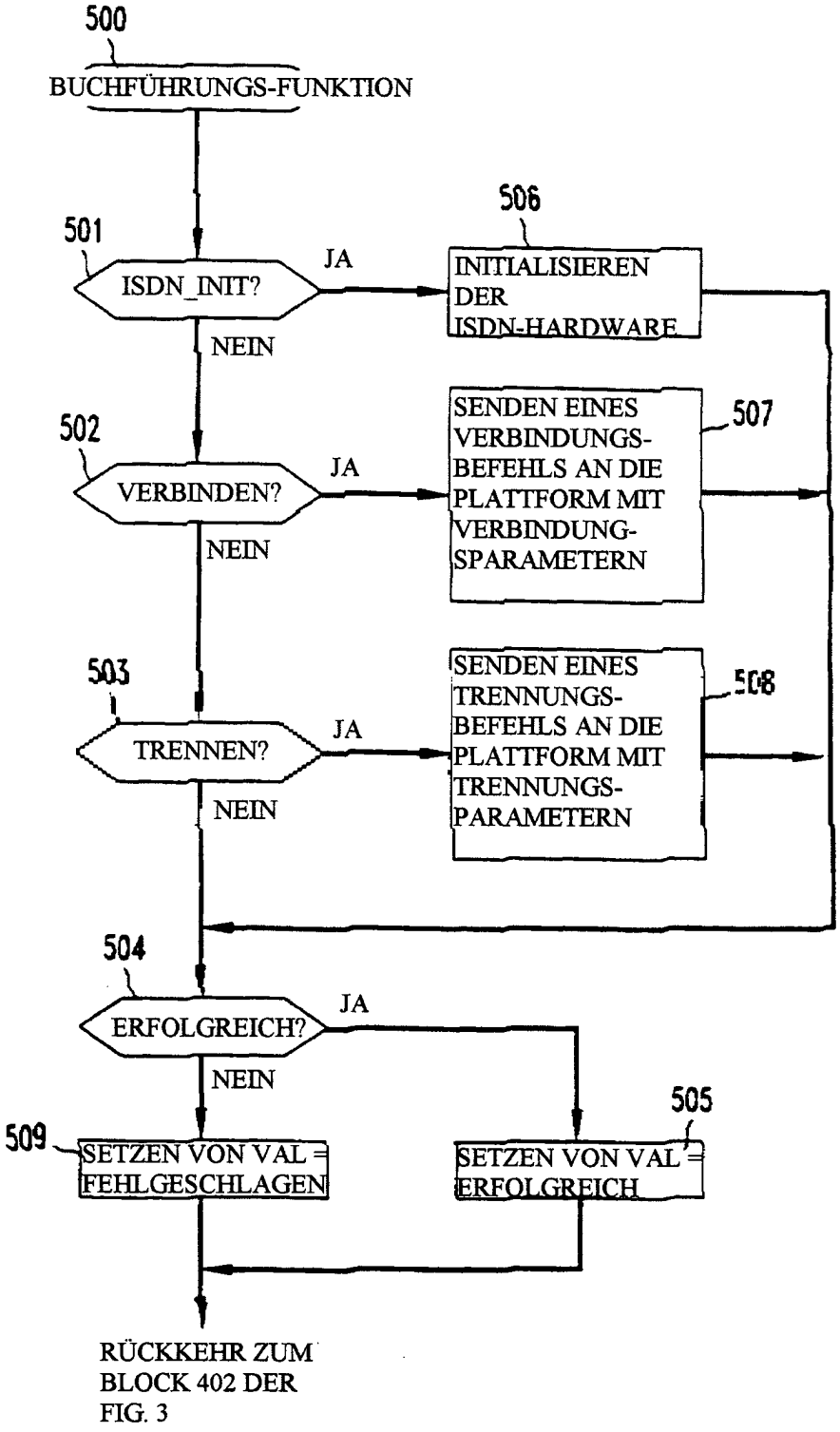

500
BUCHFÜHRUNGS-FUNKTION
501
ISDN_INIT?
JA
NEIN
506
INITIALISIEREN DER ISDN-HARDWARE
502
VERBINDEN?
JA
NEIN
SENDEN EINES VERBINDUNGS-BEFEHLS AN DIE PLATTFORM MIT VERBINDUNG-SPARAMETERN
507
503
TRENNEN?
JA
NEIN
SENDEN EINES TRENNUNGS-BEFEHLS AN DIE PLATTFORM MIT TRENNUNGS-PARAMETERN
508
504
ERFOLGREICH?
JA
NEIN
505
509
SETZEN VON VAL = FEHLGESCHLAGEN
SETZEN VON VAL = ERFOLGREICH
RÜCKKEHR ZUM BLOCK 402 DER FIG. 3

FIG. 5

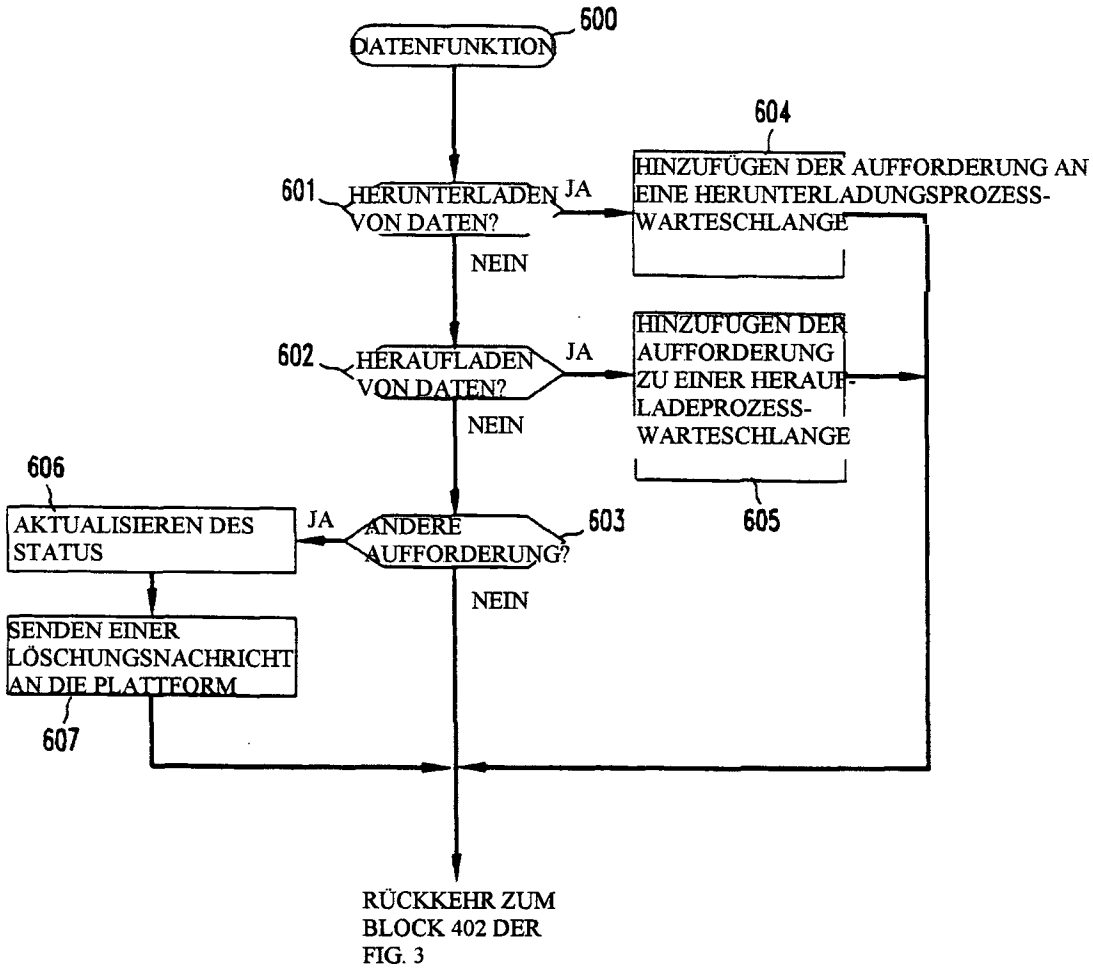

600
DATENFUNKTION
604
HINZUFÜGEN DER AUFFORDERUNG AN EINE HERUNTERLADUNGSPROZESS-WARTESCHLANGE
601
HERUNTERLADEN VON DATEN?
JA
NEIN
602
HERAUFLADEN VON DATEN?
JA
NEIN
HINZUFUGEN DER AUFFORDERUNG ZU EINER HERAUF-LADEPROZESS-WARTESCHLANGE
605
606
AKTUALISIEREN DES STATUS
JA
ANDERE AUFFORDERUNG?
603
NEIN
SENDEN EINER LÖSCHUNGSNACHRICHT AN DIE PLATTFORM
607
RÜCKKEHR ZUM BLOCK 402 DER FIG. 3

FIG. 6

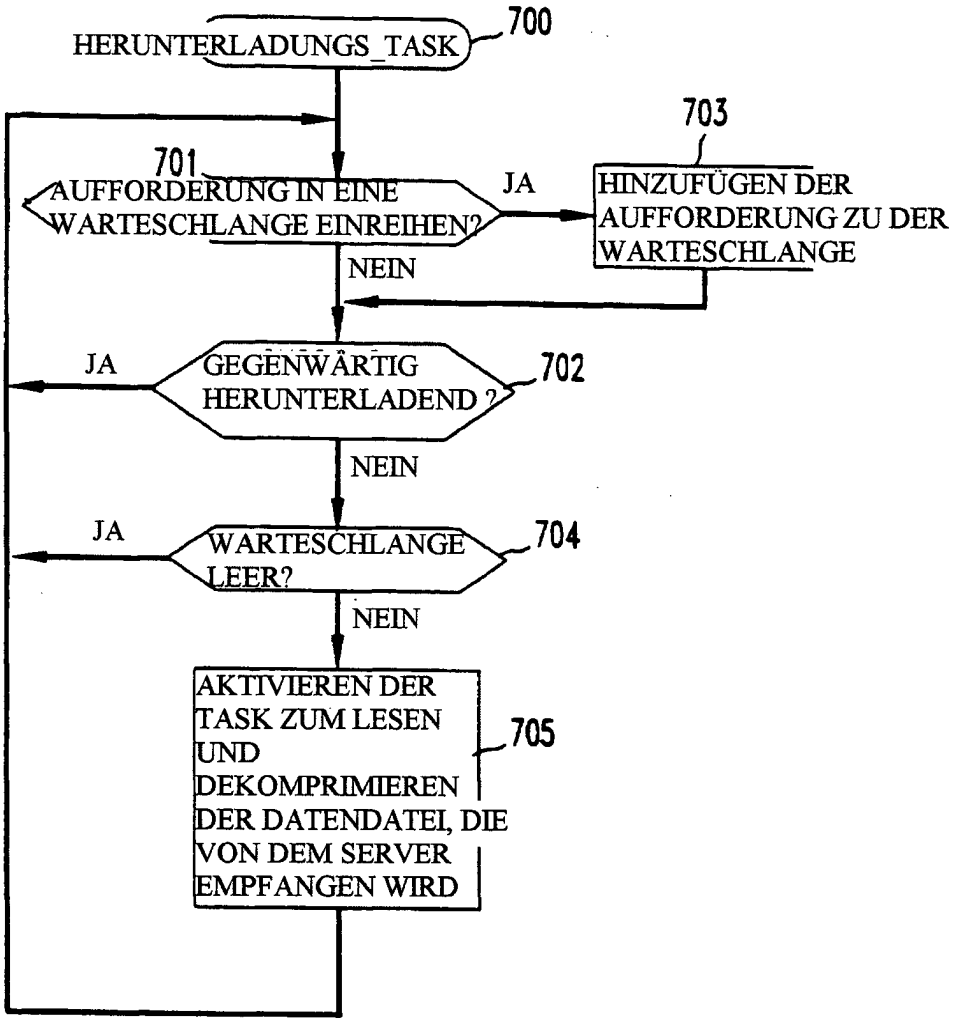

HERUNTERLADUNGS_TASK
700
703
701
AUFFORDERUNG IN EINE WARTESCHLANGE EINREIHEN?
JA
HINZUFÜGEN DER AUFFORDERUNG ZU DER WARTESCHLANGE
NEIN
JA
GEGENWÄRTIG HERUNTERLADEND ?
702
NEIN
JA
WARTESCHLANGE LEER?
704
NEIN
AKTIVIEREN DER TASK ZUM LESEN UND DEKOMPRIMIEREN DER DATENDATEI, DIE VON DEM SERVER EMPFANGEN WIRD
705

FIG. 7

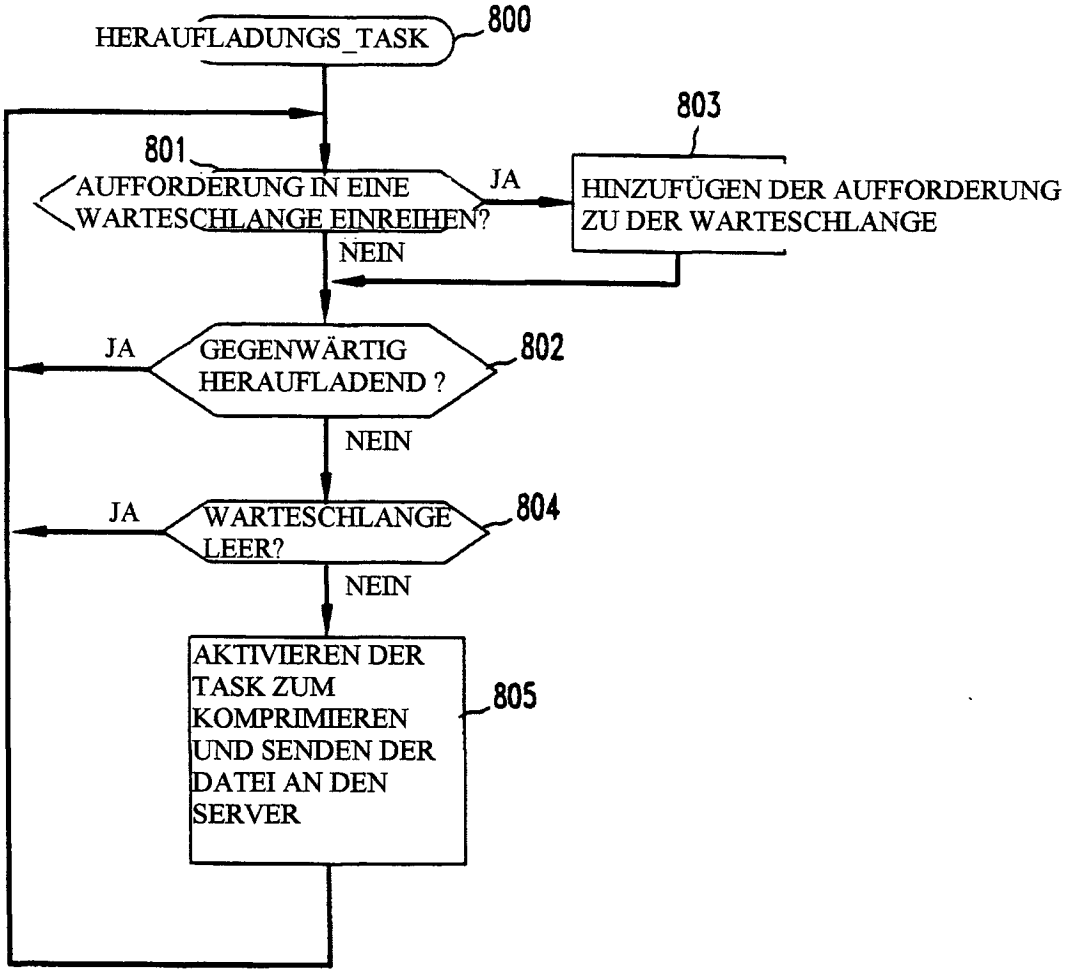
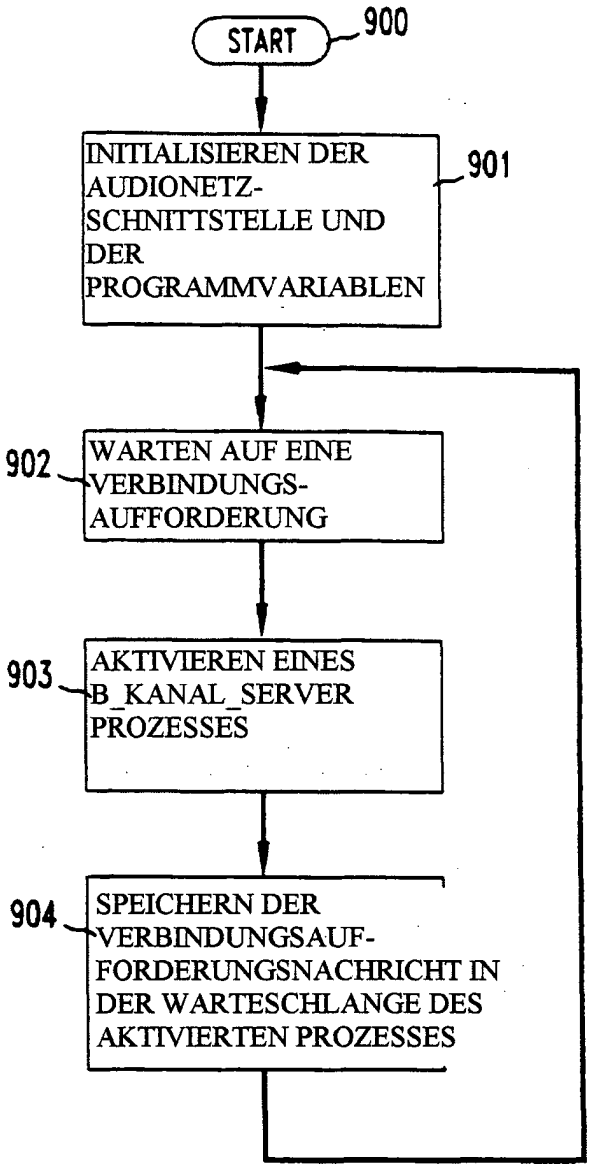

HERAUFLADUNGS_TASK
800
803
801
AUFFORDERUNG IN EINE WARTESCHLANGE EINREIHEN?
JA
HINZUFÜGEN DER AUFFORDERUNG ZU DER WARTESCHLANGE
NEIN
JA
GEGENWÄRTIG HERAUFLADEND ?
802
NEIN
JA
WARTESCHLANGE LEER?
804
NEIN
AKTIVIEREN DER TASK ZUM KOMPRIMIEREN UND SENDEN DER DATEI AN DEN SERVER
805

FIG. 8


START
900
INITIALISIEREN DER AUDIONETZ-SCHNITTSTELLE UND DER PROGRAMMVARIABLEN
901
WARTEN AUF EINE VERBINDUNGS-AUFFORDERUNG
902
AKTIVIEREN EINES B_KANAL_SERVER PROZESSES
903
SPEICHERN DER VERBINDUNGSAUF-FORDERUNGSNACHRICHT IN DER WARTESCHLANGE DES AKTIVIERTEN PROZESSES
904

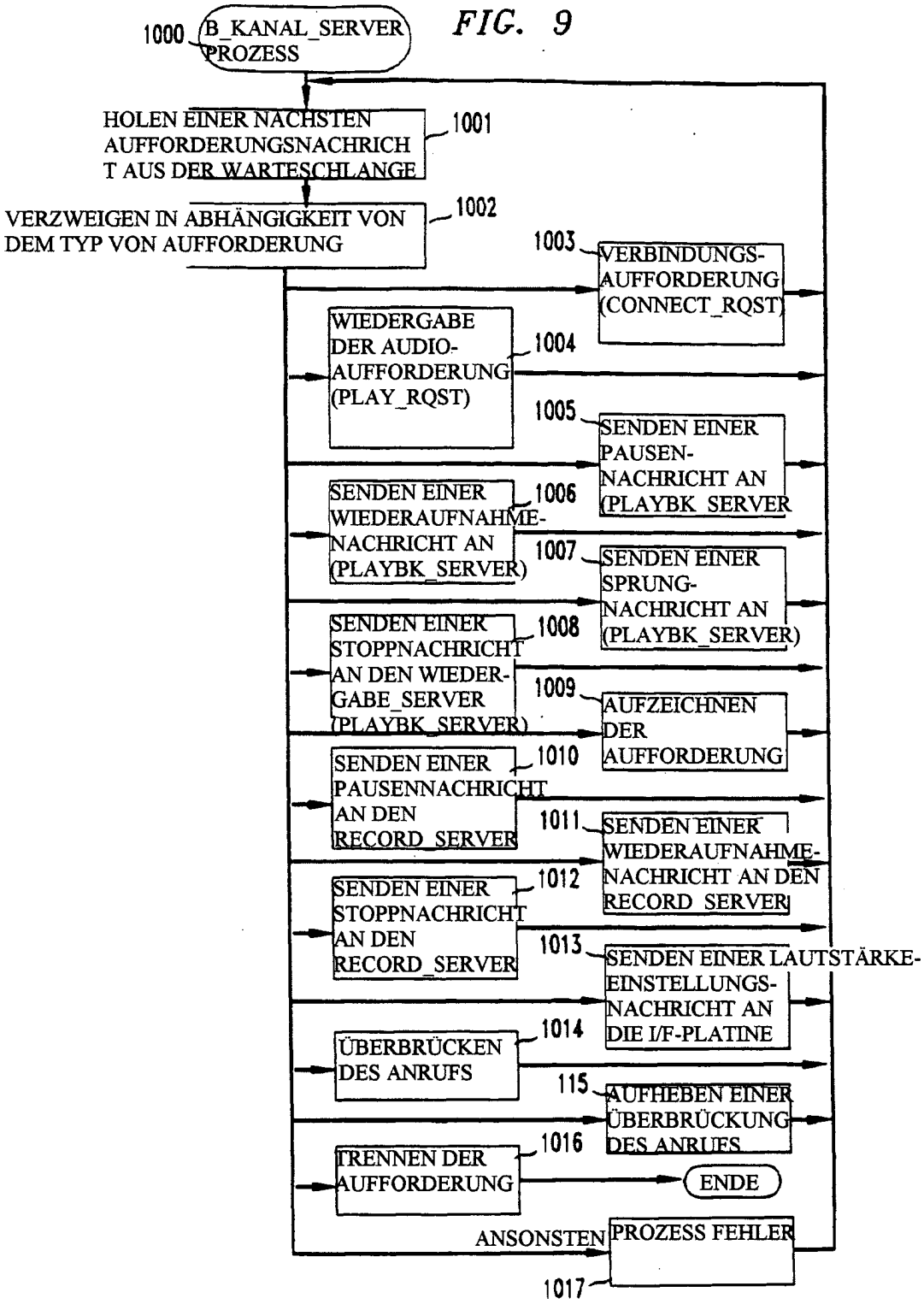

1000
B_KANAL_SERVER PROZESS
HOLEN EINER NACHSTEN AUFFORDERUNGSNACHRICHT AUS DER WARTESCHLANGE
1001
VERZWEIGEN IN ABHÄNGIGKEIT VON DEM TYP VON AUFFORDERUNG
1002
1003
VERBINDUNGS-AUFFORDERUNG (CONNECT_RQST)
WIEDERGABE DER AUDIO-AUFFORDERUNG (PLAY_RQST)
1004
1005
SENDEN EINER PAUSEN-NACHRICHT AN (PLAYBK_SERVER
SENDEN EINER WIEDERAUFNAHME-NACHRICHT AN (PLAYBK_SERVER)
1006
1007
SENDEN EINER SPRUNG-NACHRICHT AN (PLAYBK_SERVER)
SENDEN EINER STOPPNACHRICHT AN DEN WIEDER-GABE_SERVER (PLAYBK_SERVER)
1008
1009
AUFZEICHNEN DER AUFFORDERUNG
SENDEN EINER PAUSENNACHRICHT AN DEN RECORD_SERVER
1010
1011
SENDEN EINER WIEDERAUFNAHME-NACHRICHT AN DEN RECORD_SERVER
SENDEN EINER STOPPNACHRICHT AN DEN RECORD_SERVER
1012
1013
SENDEN EINER LAUTSTÄRKE-EINSTELLUNGS-NACHRICHT AN DIE I/F-PLATINE
ÜBERBRÜCKEN DES ANRUFS
1014
115
AUFHEBEN EINER ÜBERBRÜCKUNG DES ANRUFS
TRENNEN DER AUFFORDERUNG
1016
ENDE
ANSONSTEN
PROZESS FEHLER
1017

FIG. 9

FIG. 10

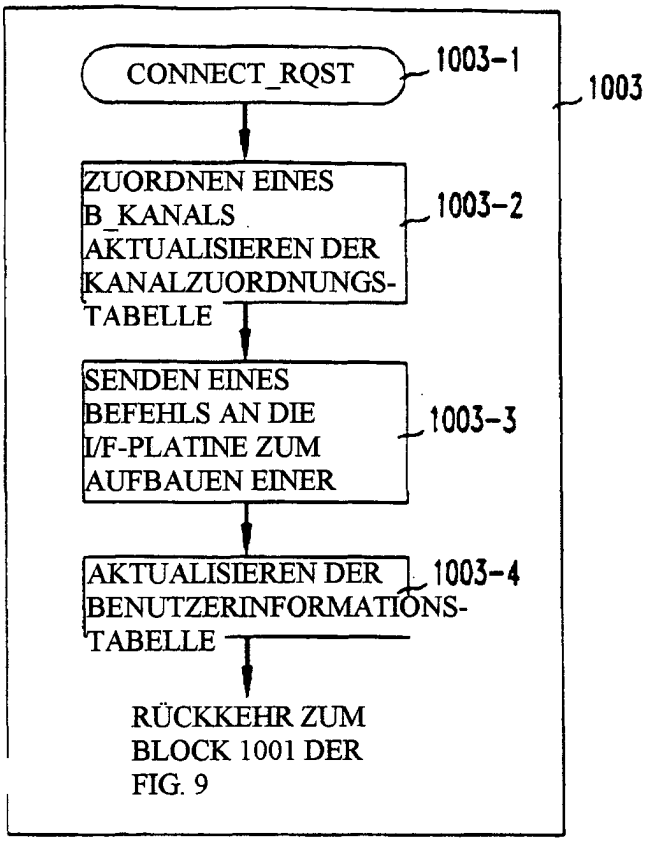

CONNECT_RQST
1003-1
1003
ZUORDNEN EINES B_KANALS AKTUALISIEREN DER KANALZUORDNUNGS-TABELLE
1003-2
SENDEN EINES BEFEHLS AN DIE I/F-PLATINE ZUM AUFBAUEN EINER
1003-3
AKTUALISIEREN DER BENUTZERINFORMATIONS-TABELLE
1003-4
RÜCKKEHR ZUM BLOCK 1001 DER FIG. 9

FIG. 11

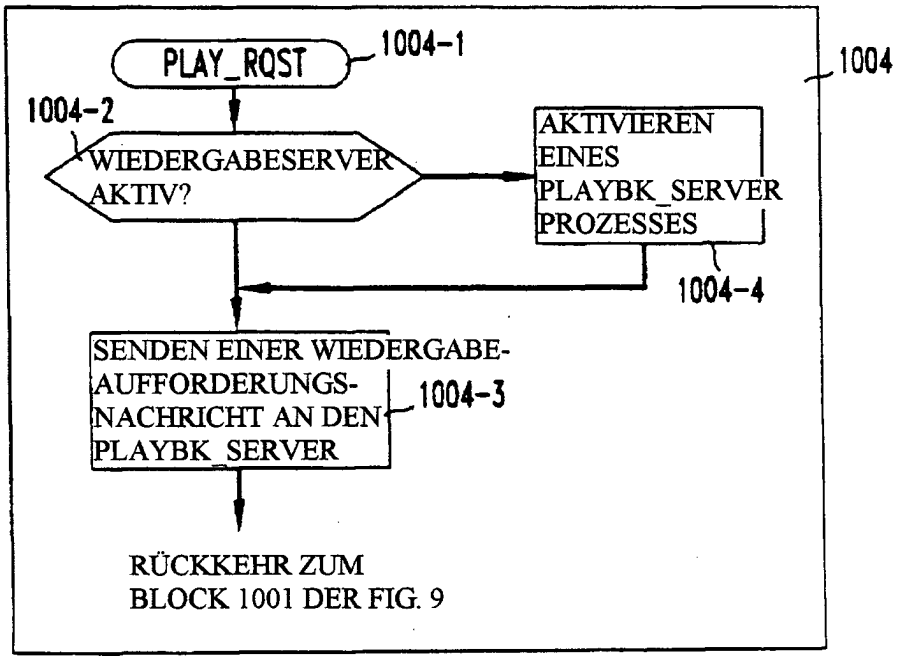

PLAY_RQST
1004-1
1004
1004-2
WIEDERGABESERVER AKTIV?
AKTIVIEREN EINES PLAYBK_SERVER PROZESSES
1004-4
SENDEN EINER WIEDERGABE-AUFFORDERUNGS-NACHRICHT AN DEN PLAYBK_SERVER
1004-3
RÜCKKEHR ZUM BLOCK 1001 DER FIG. 9

FIG. 12

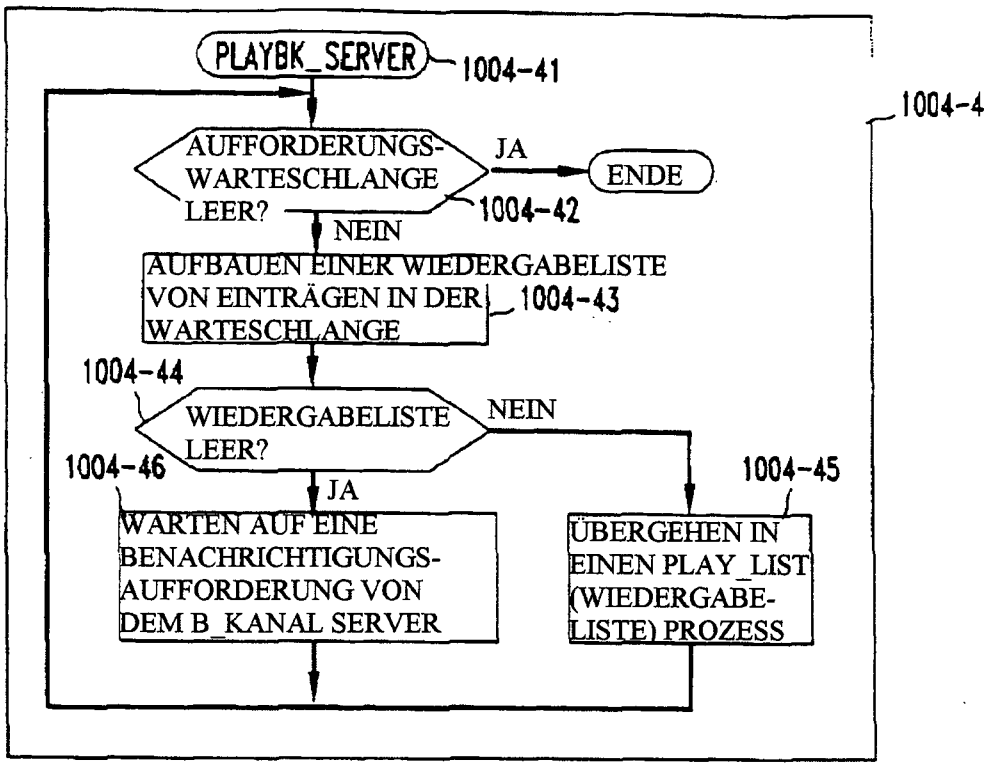

PLAYBK_SERVER
1004-41
1004-4
AUFFORDERUNGS-WARTESCHLANGE LEER?
JA
ENDE
1004-42
NEIN
AUFBAUEN EINER WIEDERGABELISTE VON EINTRÄGEN IN DER WARTESCHLANGE
1004-43
1004-44
WIEDERGABELISTE LEER?
NEIN
1004-46
JA
1004-45
WARTEN AUF EINE BENACHRICHTIGUNGS-AUFFORDERUNG VON DEM B_KANAL SERVER
ÜBERGEHEN IN EINEN PLAY_LIST (WIEDERGABE-LISTE) PROZESS

FIG. 13

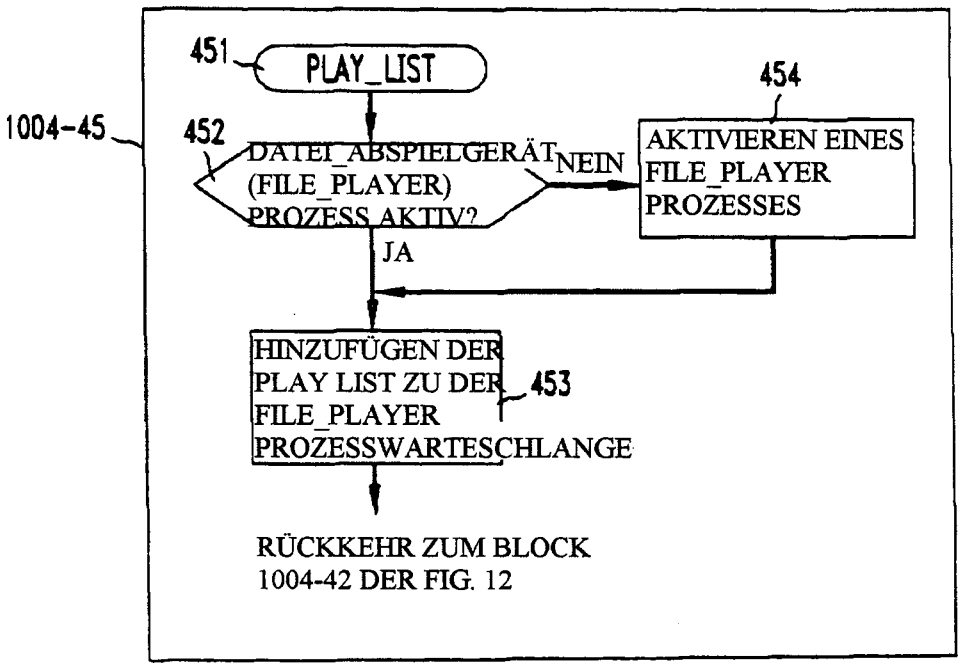

451
PLAY_LIST
1004-45
454
452
DATEI_ABSPIELGERÄT (FILE_PLAYER) PROZESS AKTIV?
NEIN
AKTIVIEREN EINES FILE_PLAYER PROZESSES
JA
HINZUFÜGEN DER PLAY LIST ZU DER FILE_PLAYER PROZESSWARTESCHLANGE
453
RÜCKKEHR ZUM BLOCK 1004-42 DER FIG. 12

FIG. 14

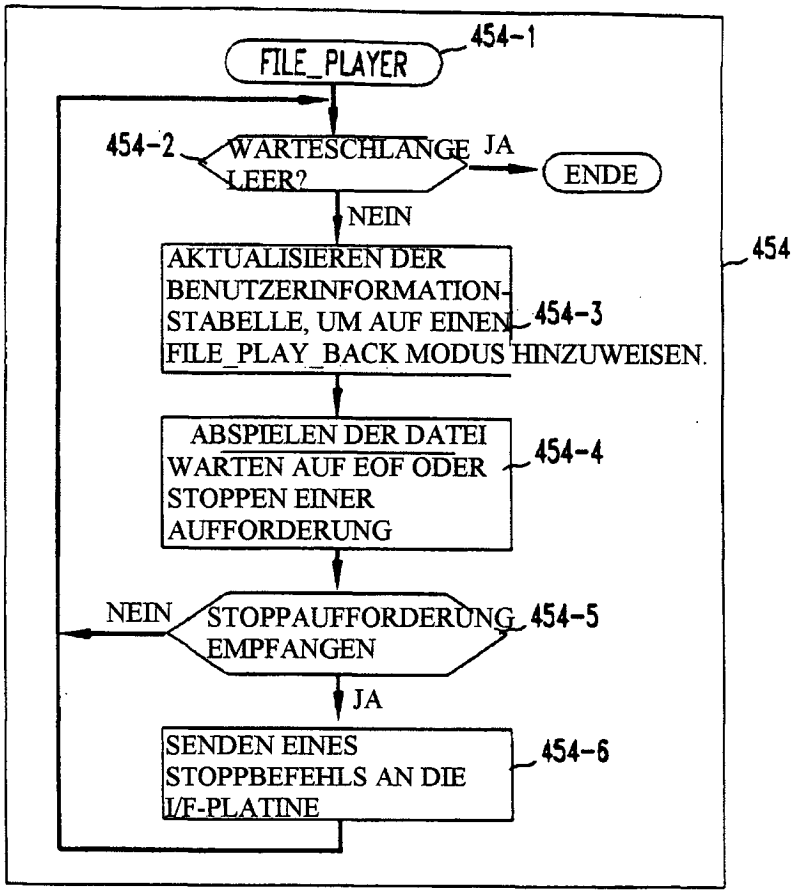

FILE_PLAYER
454-1
454-2
WARTESCHLANGE LEER?
JA
ENDE
NEIN
AKTUALISIEREN DER BENUTZERINFORMATION-STABELLE, UM AUF EINEN FILE_PLAY_BACK MODUS HINZUWEISEN.
454-3
ABSPIELEN DER DATEI WARTEN AUF EOF ODER STOPPEN EINER AUFFORDERUNG
454-4
NEIN
STOPPAUFFORDERUNG EMPFANGEN
454-5
JA
SENDEN EINES STOPPBEFEHLS AN DIE I/F-PLATINE
454-6
454

FIG. 15

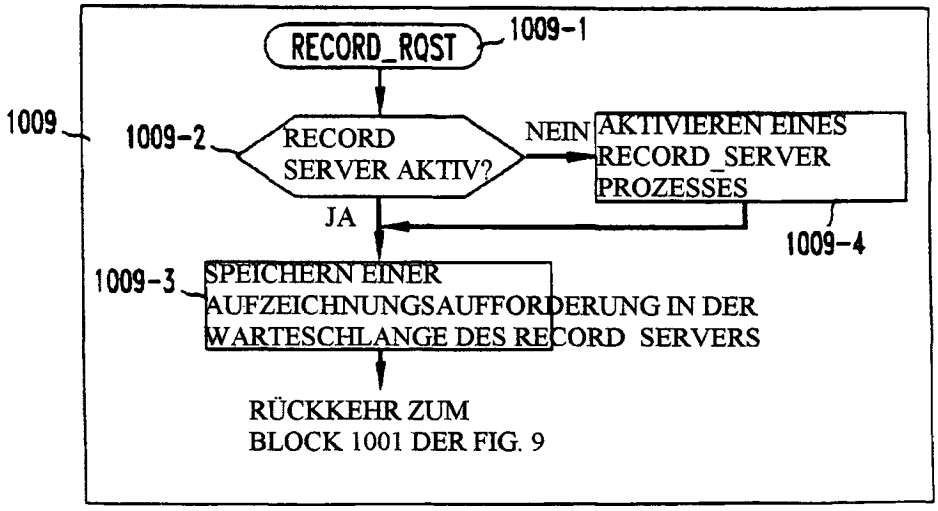

RECORD_RQST
1009-1
1009
1009-2
RECORD SERVER AKTIV?
NEIN
AKTIVIEREN EINES RECORD_SERVER PROZESSES
1009-4
JA
1009-3
SPEICHERN EINER AUFZEICHNUNGSAUFFORDERUNG IN DER WARTESCHLANGE DES RECORD SERVERS
RÜCKKEHR ZUM BLOCK 1001 DER FIG. 9

FIG. 16

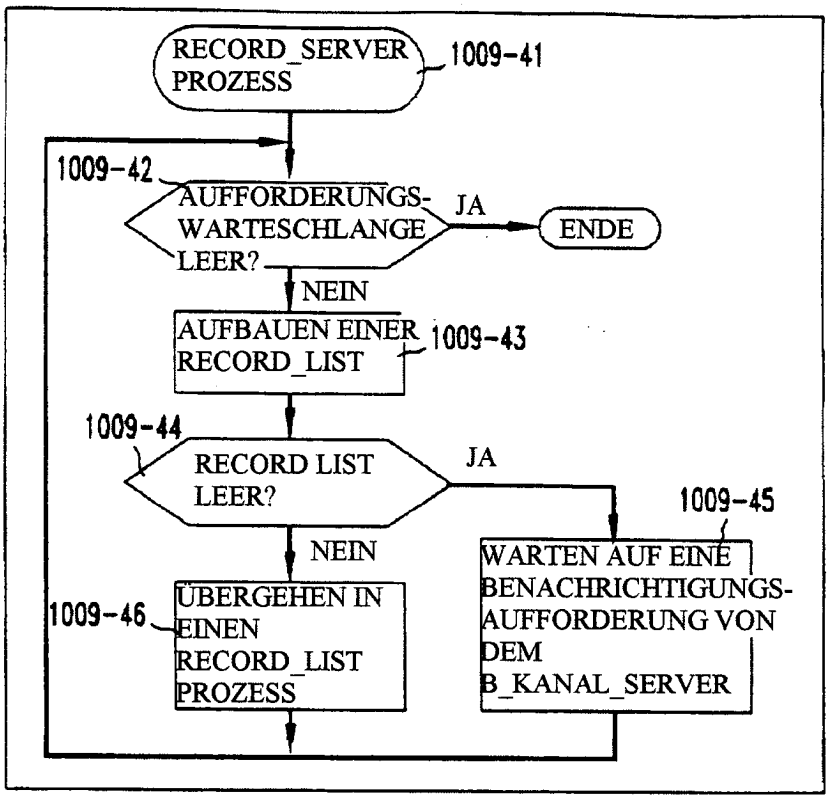

RECORD_SERVER PROZESS
1009-41
1009-42
AUFFORDERUNGS-WARTESCHLANGE LEER?
JA
ENDE
NEIN
AUFBAUEN EINER RECORD_LIST
1009-43
1009-44
RECORD LIST LEER?
JA
1009-45
NEIN
WARTEN AUF EINE BENACHRICHTIGUNGS-AUFFORDERUNG VON DEM B_KANAL_SERVER
1009-46
ÜBERGEHEN IN EINEN RECORD_LIST PROZESS

FIG. 17

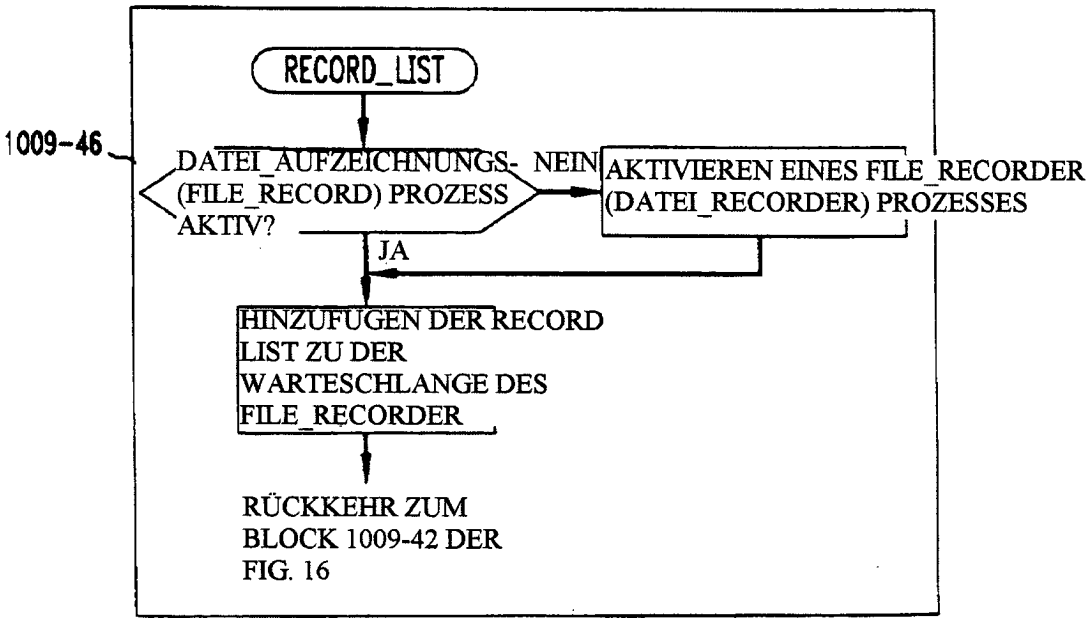

RECORD_LIST
1009-46
DATEI_AUFZEICHNUNGS-(FILE_RECORD) PROZESS AKTIV?
NEIN
AKTIVIEREN EINES FILE_RECORDER (DATEI_RECORDER) PROZESSES
JA
HINZUFÜGEN DER RECORD LIST ZU DER WARTESCHLANGE DES FILE_RECORDER
RÜCKKEHR ZUM BLOCK 1009-42 DER FIG. 16

FIG. 18

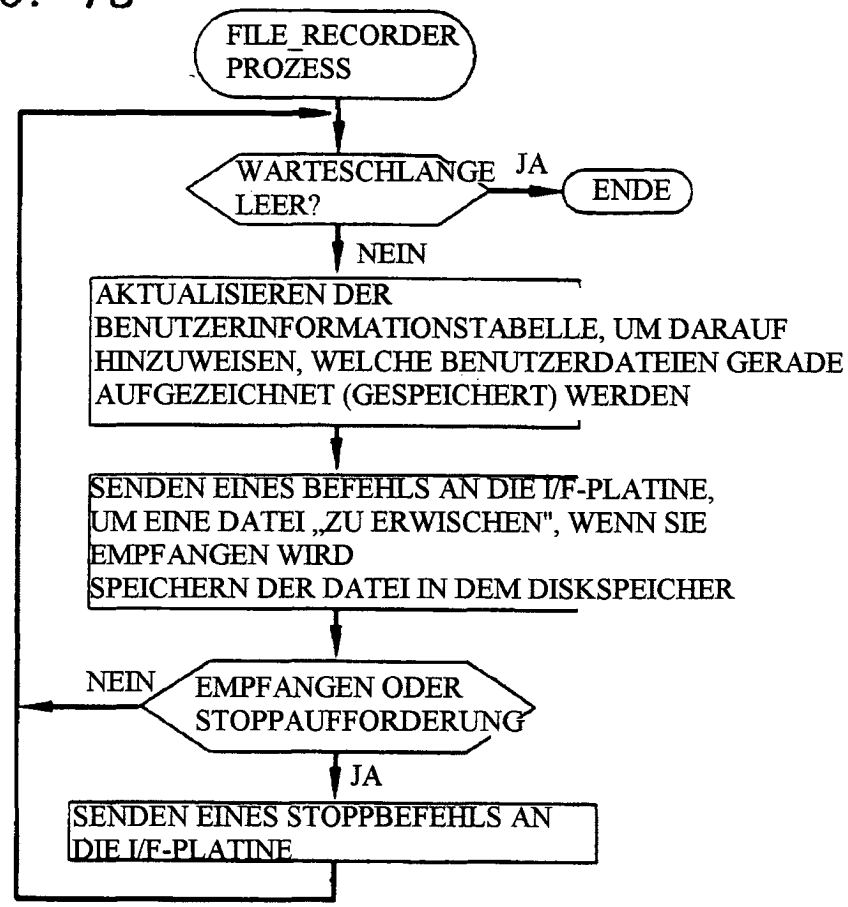

FILE_RECORDER PROZESS
WARTESCHLANGE LEER?
JA
ENDE
NEIN
AKTUALISIEREN DER BENUTZERINFORMATIONSTABELLE, UM DARAUF HINZUWEISEN, WELCHE BENUTZERDATEIEN GERADE AUFGEZEICHNET (GESPEICHERT) WERDEN
SENDEN EINES BEFEHLS AN DIE I/F-PLATINE, UM EINE DATEI „ZU ERWISCHEN", WENN SIE EMPFANGEN WIRD
SPEICHERN DER DATEI IN DEM DISKSPEICHER
NEIN
EMPFANGEN ODER STOPPAUFFORDERUNG
JA
SENDEN EINES STOPPBEFEHLS AN DIE I/F-PLATINE

FIG. 19

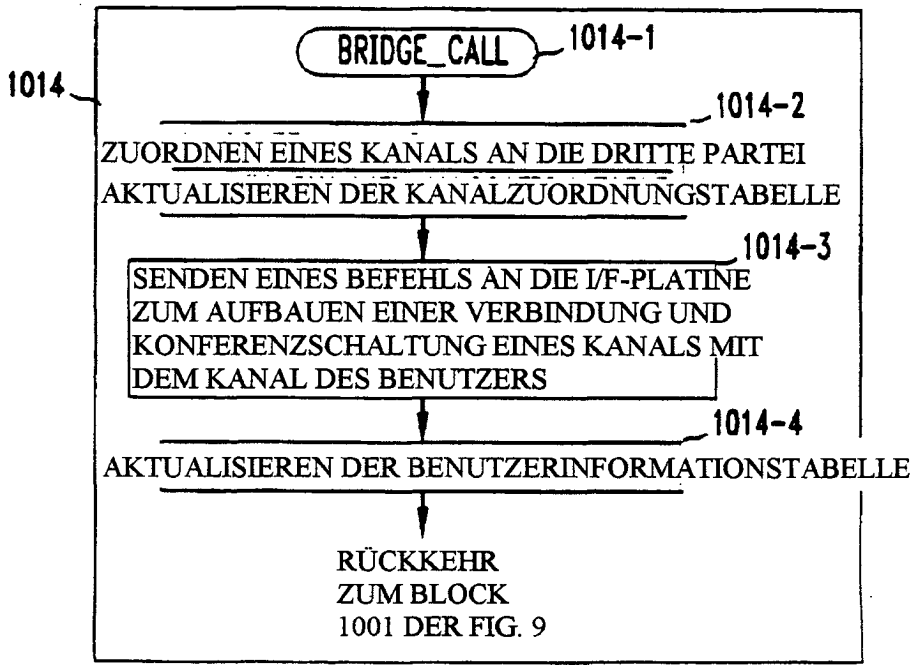

1014
BRIDGE_CALL
1014-1
1014-2
ZUORDNEN EINES KANALS AN DIE DRITTE PARTEI
AKTUALISIEREN DER KANALZUORDNUNGSTABELLE
1014-3
SENDEN EINES BEFEHLS AN DIE I/F-PLATINE ZUM AUFBAUEN EINER VERBINDUNG UND KONFERENZSCHALTUNG EINES KANALS MIT DEM KANAL DES BENUTZERS
1014-4
AKTUALISIEREN DER BENUTZERINFORMATIONSTABELLE
RÜCKKEHR ZUM BLOCK 1001 DER FIG. 9

FIG. 20

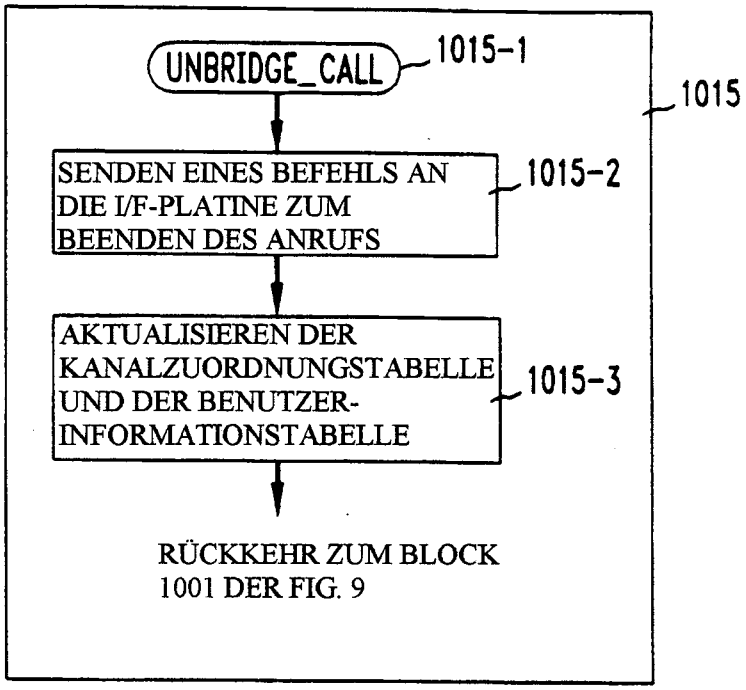

UNBRIDGE_CALL
1015-1
1015
SENDEN EINES BEFEHLS AN DIE I/F-PLATINE ZUM BEENDEN DES ANRUFS
1015-2
AKTUALISIEREN DER KANALZUORDNUNGSTABELLE UND DER BENUTZER-INFORMATIONSTABELLE
1015-3
RÜCKKEHR ZUM BLOCK 1001 DER FIG. 9

FIG. 21

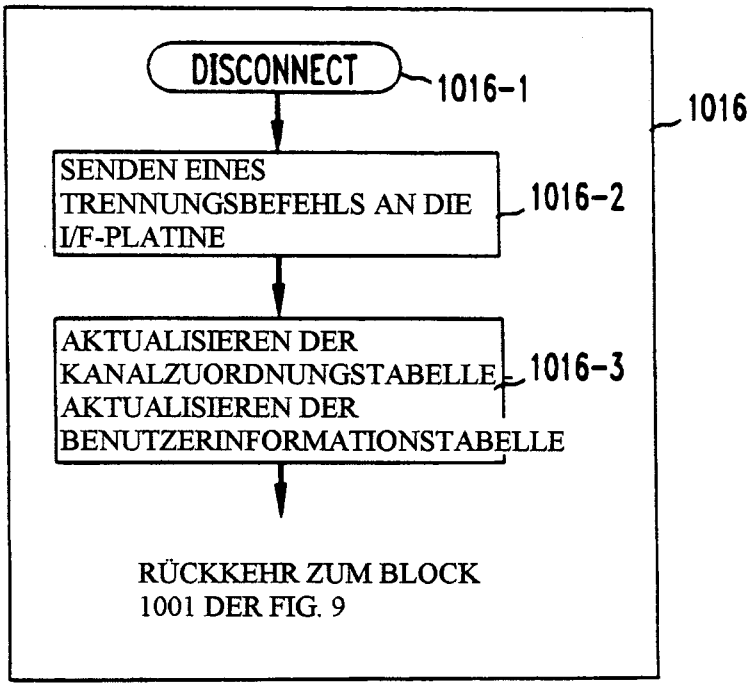

DISCONNECT
1016-1
1016
SENDEN EINES TRENNUNGSBEFEHLS AN DIE I/F-PLATINE
1016-2
AKTUALISIEREN DER KANALZUORDNUNGSTABELLE
AKTUALISIEREN DER BENUTZERINFORMATIONSTABELLE
1016-3
RÜCKKEHR ZUM BLOCK 1001 DER FIG. 9

FIG. 23

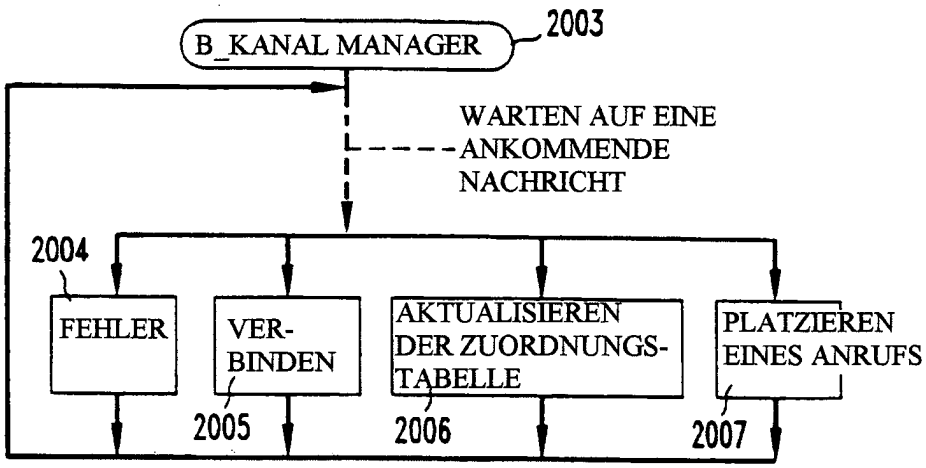

B_KANAL MANAGER
2003
WARTEN AUF EINE ANKOMMENDE NACHRICHT
2004
FEHLER
VER-BINDEN
AKTUALISIEREN DER ZUORDNUNGS-TABELLE
PLATZIEREN EINES ANRUFS
2005
2006
2007

FIG. 24

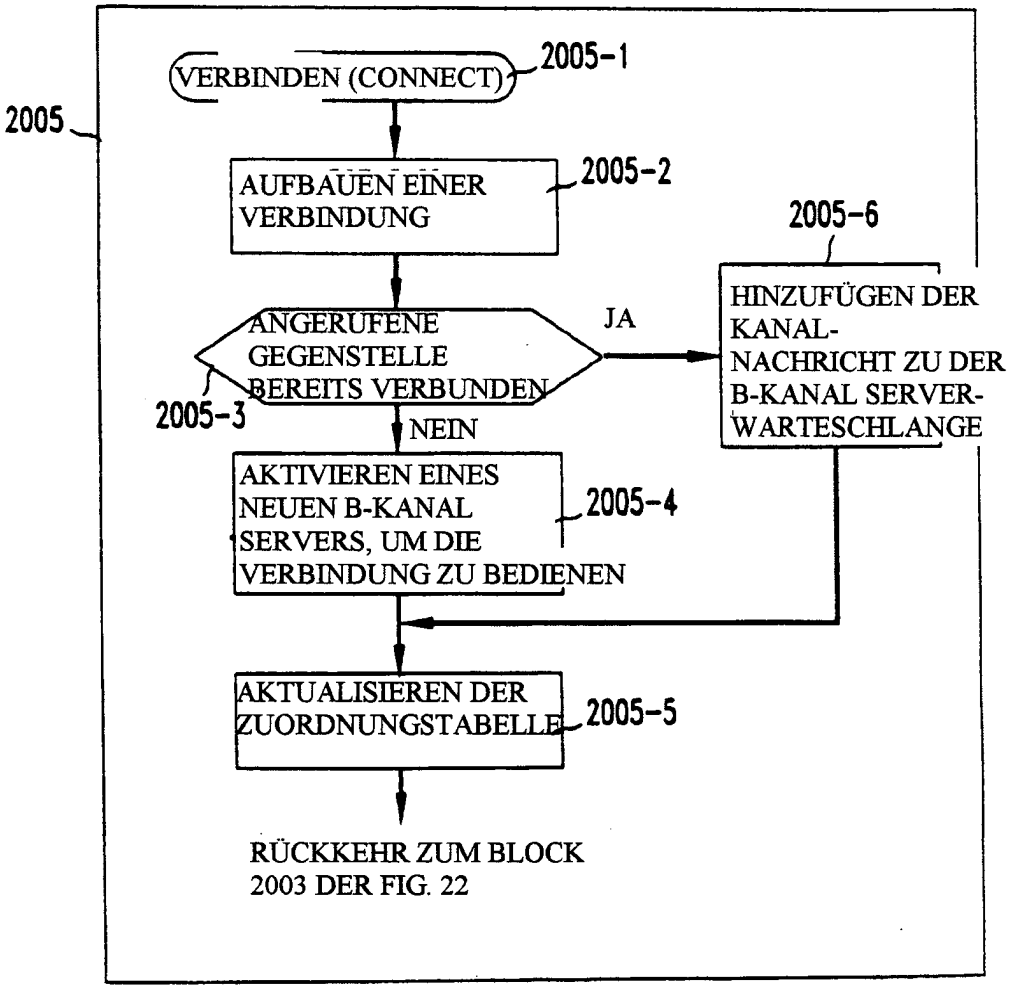

2005
VERBINDEN (CONNECT)
2005-1
AUFBAUEN EINER VERBINDUNG
2005-2
ANGERUFENE GEGENSTELLE BEREITS VERBUNDEN
2005-3
JA
NEIN
2005-6
HINZUFÜGEN DER KANAL-NACHRICHT ZU DER B-KANAL SERVER-WARTESCHLANGE
AKTIVIEREN EINES NEUEN B-KANAL SERVERS, UM DIE VERBINDUNG ZU BEDIENEN
2005-4
AKTUALISIEREN DER ZUORDNUNGSTABELLE
2005-5
RÜCKKEHR ZUM BLOCK 2003 DER FIG. 22

FIG. 25

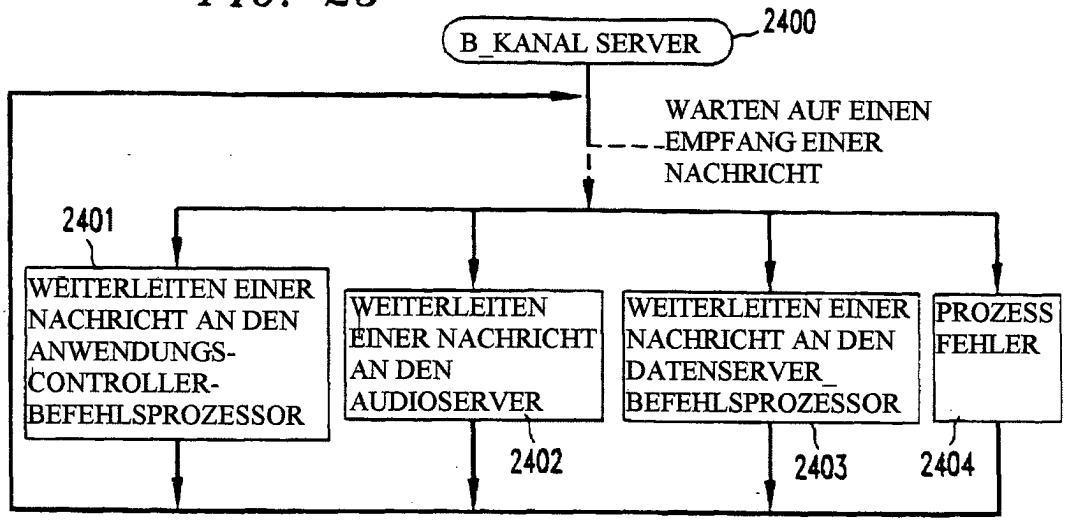

B_KANAL SERVER
2400
WARTEN AUF EINEN EMPFANG EINER NACHRICHT
2401
WEITERLEITEN EINER NACHRICHT AN DEN ANWENDUNGS-CONTROLLER-BEFEHLSPROZESSOR
WEITERLEITEN EINER NACHRICHT AN DEN AUDIOSERVER
2402
WEITERLEITEN EINER NACHRICHT AN DEN DATENSERVER_ BEFEHLSPROZESSOR
2403
PROZESS FEHLER
2404

FIG. 31

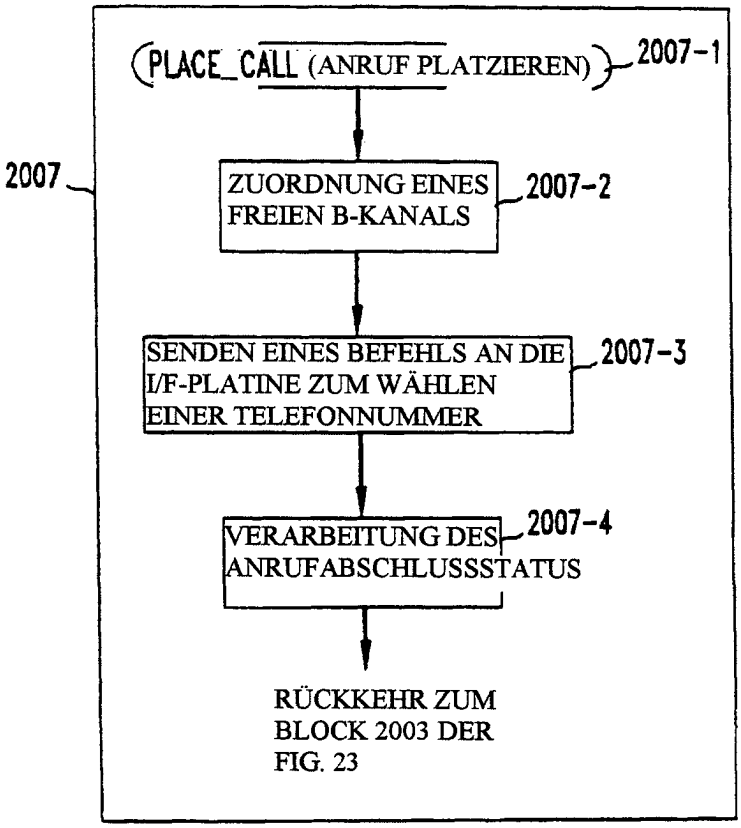

PLACE_CALL (ANRUF PLATZIEREN)
2007-1
2007
ZUORDNUNG EINES FREIEN B-KANALS
2007-2
SENDEN EINES BEFEHLS AN DIE I/F-PLATINE ZUM WÄHLEN EINER TELEFONNUMMER
2007-3
VERARBEITUNG DES ANRUFABSCHLUSSSTATUS
2007-4
RÜCKKEHR ZUM BLOCK 2003 DER FIG. 23

FIG. 26

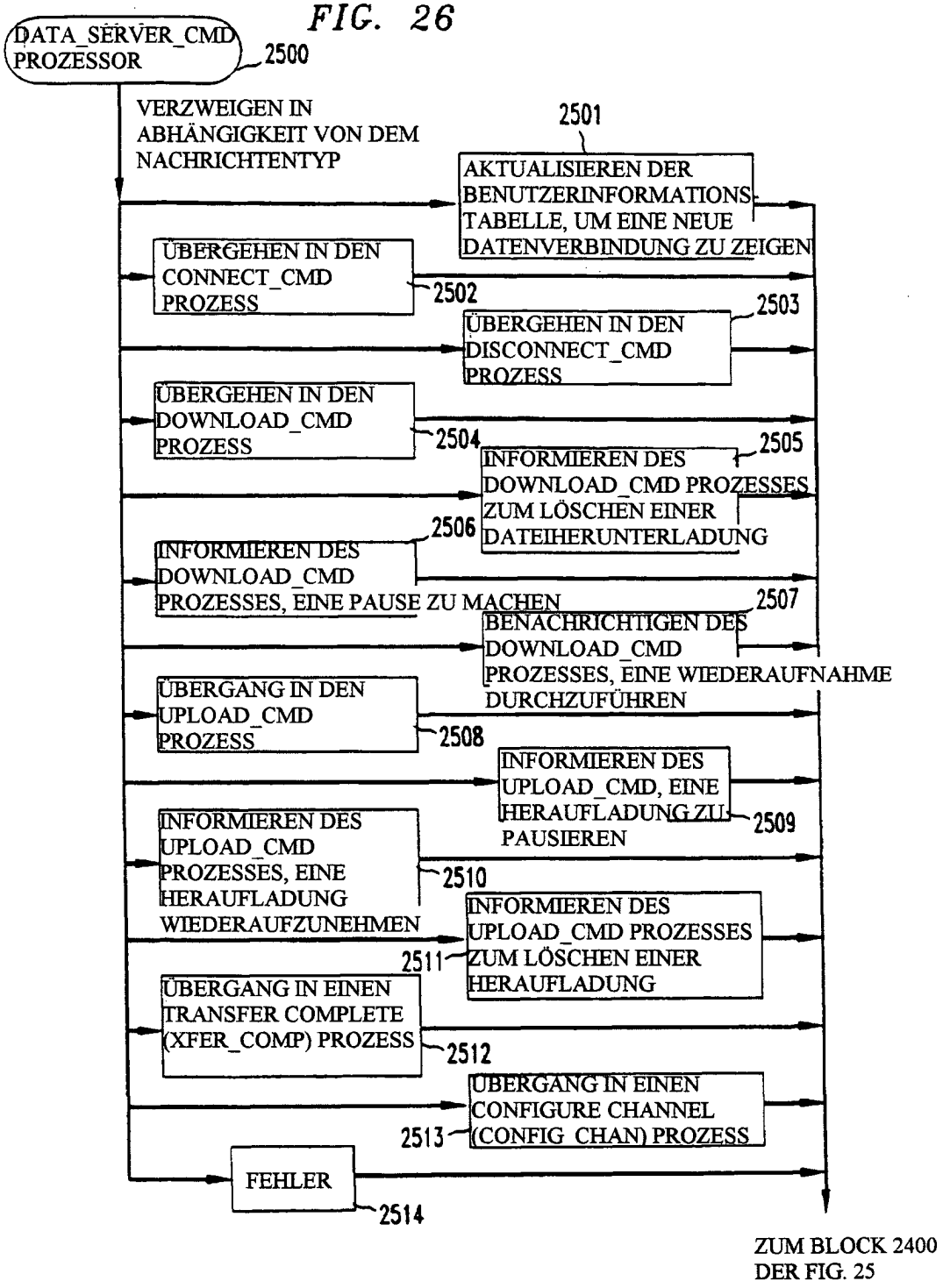

DATA_SERVER_CMD PROZESSOR
2500
VERZWEIGEN IN ABHÄNGIGKEIT VON DEM NACHRICHTENTYP
2501
AKTUALISIEREN DER BENUTZERINFORMATIONS-TABELLE, UM EINE NEUE DATENVERBINDUNG ZU ZEIGEN
ÜBERGEHEN IN DEN CONNECT_CMD PROZESS
2502
2503
ÜBERGEHEN IN DEN DISCONNECT_CMD PROZESS
ÜBERGEHEN IN DEN DOWNLOAD_CMD PROZESS
2504
2505
INFORMIEREN DES DOWNLOAD_CMD PROZESSES ZUM LÖSCHEN EINER DATEIHERUNTERLADUNG
2506
INFORMIEREN DES DOWNLOAD_CMD PROZESSES, EINE PAUSE ZU MACHEN
2507
BENACHRICHTIGEN DES DOWNLOAD_CMD PROZESSES, EINE WIEDERAUFNAHME DURCHZUFÜHREN
ÜBERGANG IN DEN UPLOAD_CMD PROZESS
2508
INFORMIEREN DES UPLOAD_CMD, EINE HERAUFLADUNG ZU PAUSIEREN
2509
INFORMIEREN DES UPLOAD_CMD PROZESSES, EINE HERAUFLADUNG WIEDERAUFZUNEHMEN
2510
INFORMIEREN DES UPLOAD_CMD PROZESSES ZUM LÖSCHEN EINER HERAUFLADUNG
2511
ÜBERGANG IN EINEN TRANSFER COMPLETE (XFER_COMP) PROZESS
2512
ÜBERGANG IN EINEN CONFIGURE CHANNEL (CONFIG_CHAN) PROZESS
2513
FEHLER
2514
ZUM BLOCK 2400 DER FIG. 25

FIG. 27

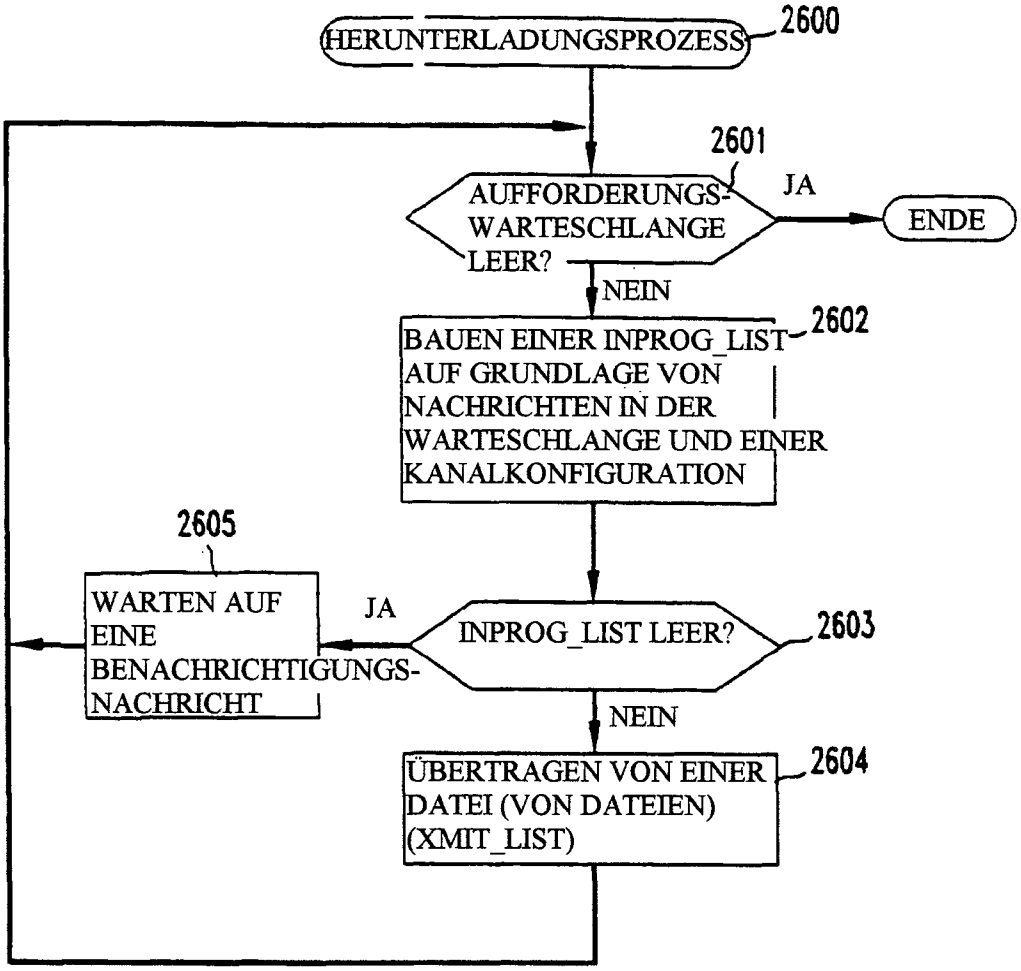

HERUNTERLADUNGSPROZESS
2600
2601
AUFFORDERUNGS-WARTESCHLANGE LEER?
JA
ENDE
NEIN
BAUEN EINER INPROG_LIST AUF GRUNDLAGE VON NACHRICHTEN IN DER WARTESCHLANGE UND EINER KANALKONFIGURATION
2602
2605
WARTEN AUF EINE BENACHRICHTIGUNGS-NACHRICHT
JA
INPROG_LIST LEER?
2603
NEIN
ÜBERTRAGEN VON EINER DATEI (VON DATEIEN) (XMIT_LIST)
2604

FIG. 22

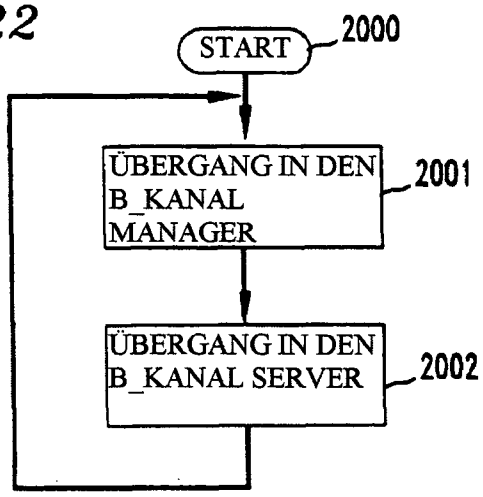

START
2000
ÜBERGANG IN DEN B_KANAL MANAGER
2001
ÜBERGANG IN DEN B_KANAL SERVER
2002

FIG. 28

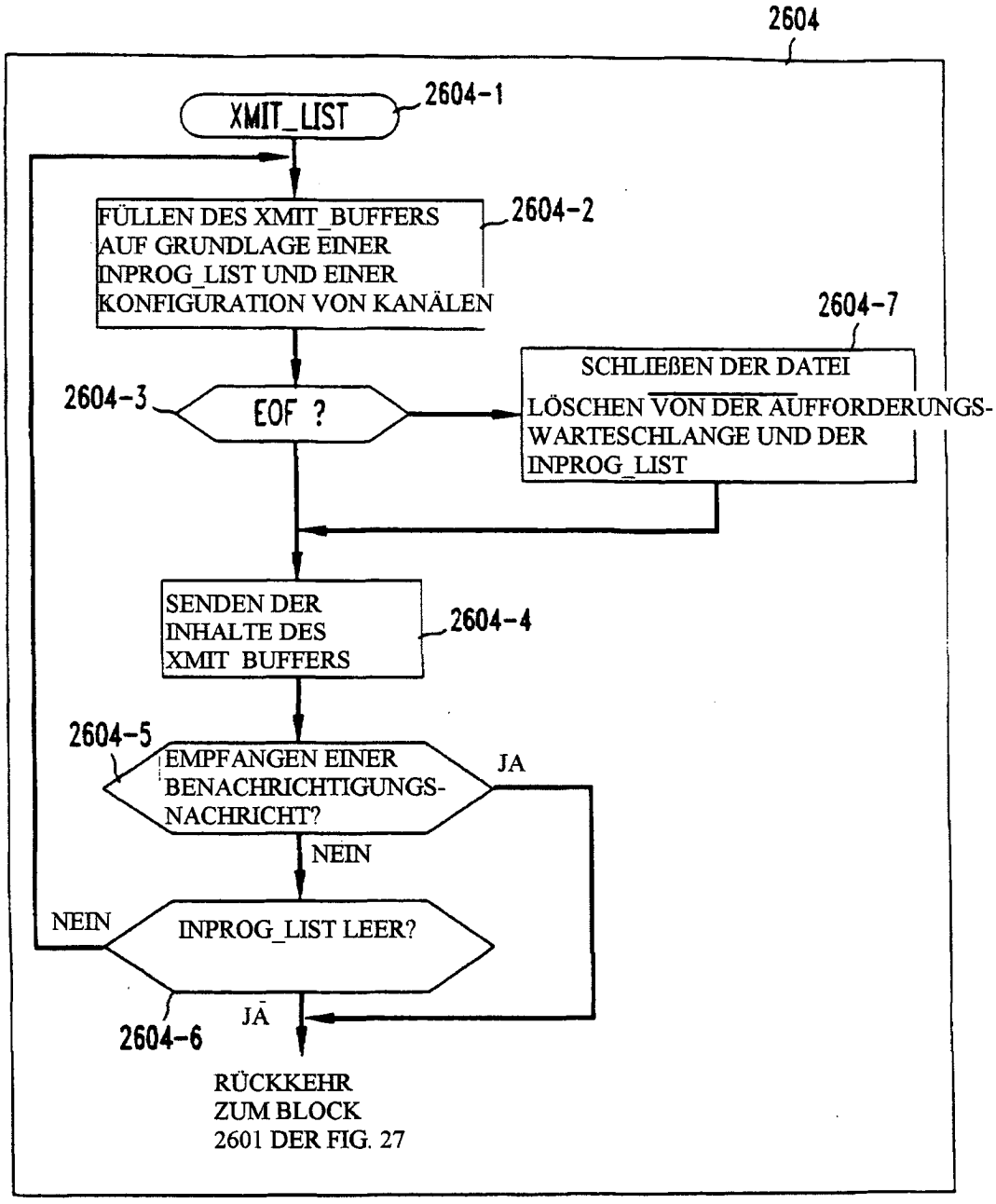

2604
XMIT_LIST
2604-1
FÜLLEN DES XMIT_BUFFERS AUF GRUNDLAGE EINER INPROG_LIST UND EINER KONFIGURATION VON KANÄLEN
2604-2
2604-7
SCHLIEßEN DER DATEI
LÖSCHEN VON DER AUFFORDERUNGS-WARTESCHLANGE UND DER INPROG_LIST
2604-3
EOF ?
SENDEN DER INHALTE DES XMIT BUFFERS
2604-4
2604-5
EMPFANGEN EINER BENACHRICHTIGUNGS-NACHRICHT?
JA
NEIN
NEIN
INPROG_LIST LEER?
JA
2604-6
RÜCKKEHR ZUM BLOCK 2601 DER FIG. 27

FIG. 29

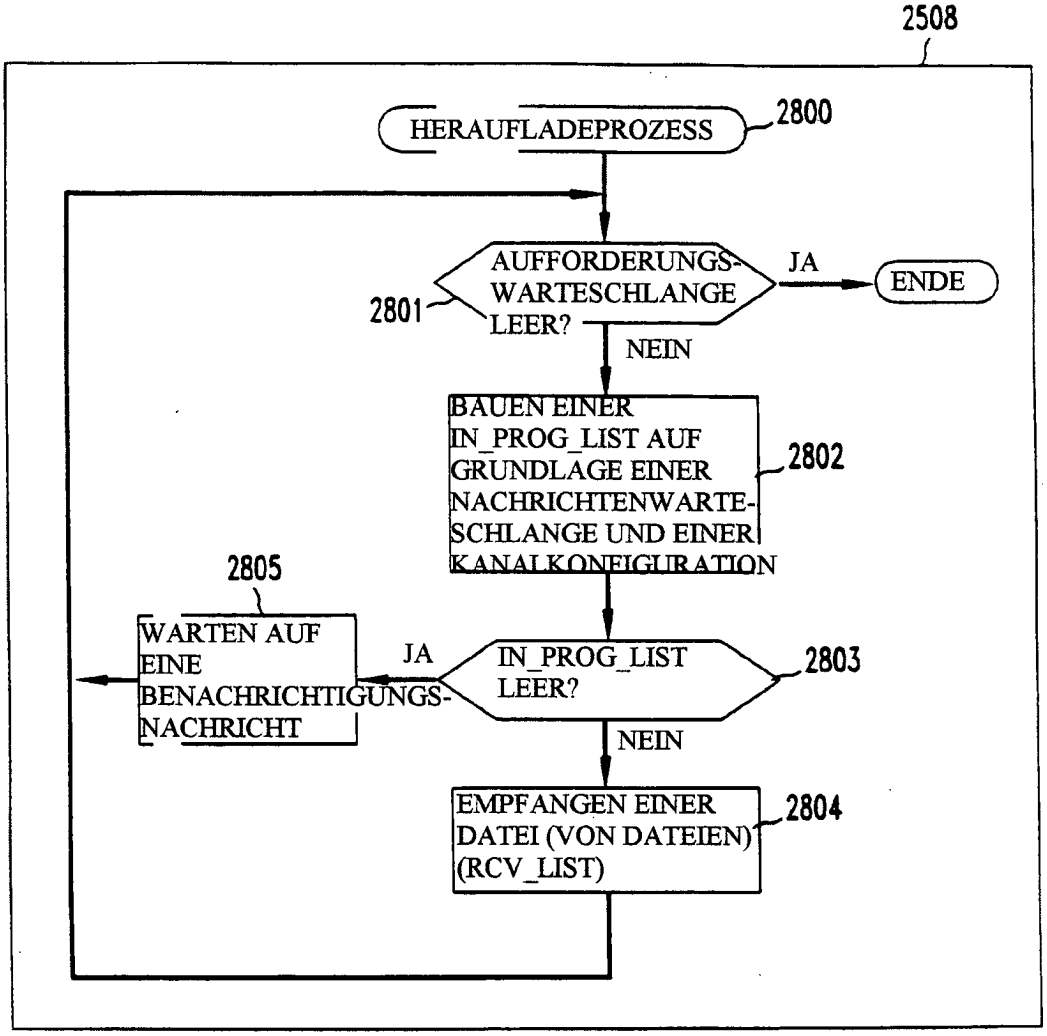

2508
HERAUFLADEPROZESS
2800
AUFFORDERUNGS-WARTESCHLANGE LEER?
2801
JA
ENDE
NEIN
BAUEN EINER IN_PROG_LIST AUF GRUNDLAGE EINER NACHRICHTENWARTE-SCHLANGE UND EINER KANALKONFIGURATION
2802
IN_PROG_LIST LEER?
2803
JA
WARTEN AUF EINE BENACHRICHTIGUNGS-NACHRICHT
2805
NEIN
EMPFANGEN EINER DATEI (VON DATEIEN) (RCV_LIST)
2804

FIG. 30

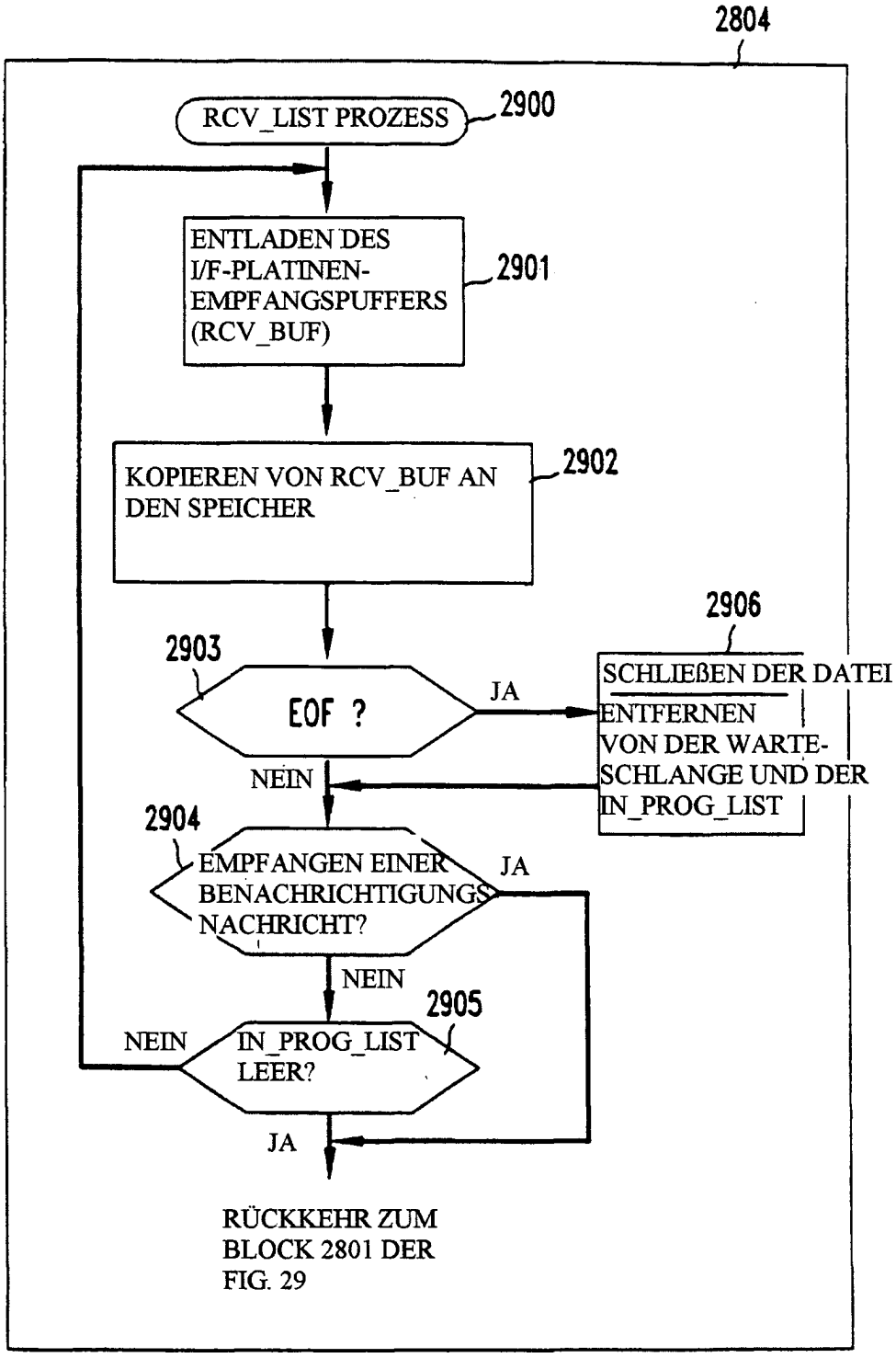
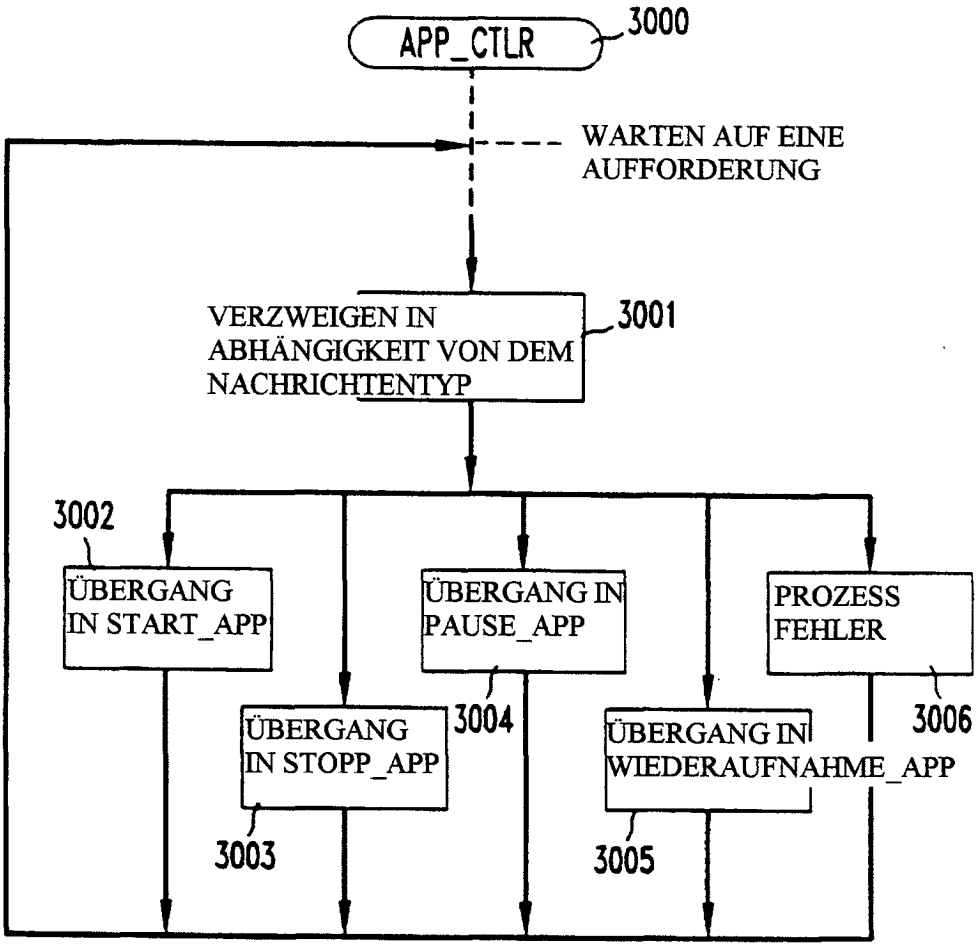

2804
RCV_LIST PROZESS
2900
ENTLADEN DES I/F-PLATINEN-EMPFANGSPUFFERS (RCV_BUF)
2901
KOPIEREN VON RCV_BUF AN DEN SPEICHER
2902
2903
EOF ?
JA
2906
SCHLIEßEN DER DATEI
ENTFERNEN VON DER WARTE-SCHLANGE UND DER IN_PROG_LIST
NEIN
2904
EMPFANGEN EINER BENACHRICHTIGUNGS NACHRICHT?
JA
NEIN
2905
NEIN
IN_PROG_LIST LEER?
JA
RÜCKKEHR ZUM BLOCK 2801 DER FIG. 29

FIG. 32


APP_CTLR
3000
WARTEN AUF EINE AUFFORDERUNG
VERZWEIGEN IN ABHÄNGIGKEIT VON DEM NACHRICHTENTYP
3001
3002
ÜBERGANG IN START_APP
ÜBERGANG IN PAUSE_APP
PROZESS FEHLER
ÜBERGANG IN STOPP_APP
3004
3006
ÜBERGANG IN WIEDERAUFNAHME_APP
3003
3005