

XAPI

- eine universelle Kommunikationsplattform -

Inhaltsverzeichnis

1. Einleitung.....	2
2. Einführung	2
3. Dienstekonzept	3
4. ISO OSI-Referenzmodell	4
4.1 Bitübertragungsschicht.....	4
4.2 Sicherungsschicht	4
4.3 Vermittlungsschicht.....	5
4.4 Transportschicht	6
4.5 Kommunikationssteuerungsschicht	6
4.6 Darstellungsschicht.....	6
4.7 Anwendungsschicht	6
5. Kommunikationsplattform	6
5.1 Zugriff auf die Bitübertragungsschicht	9
5.2 Zugriff auf die Transportschicht.....	10
5.3 Zugriff auf die Kommunikationssteuerungsschicht.....	10
5.4 Zugriff auf die Anwendungsschicht.....	10
6. XAPI	10
6.1 Dienste Endpunkt	11
6.2 Protokoll Module	11
6.3 Protokoll Stack.....	11
6.4 Service Primitive.....	12
6.5 Service Provider.....	12
6.6 Prozeß Endpunkt.....	12
6.7 Synchroner und Asynchroner Mode	12
7. Phasen der Datenkommunikation.....	13
8. Status Modell.....	14
9. Service Provider.....	16
9.1 FAX4 und BFT	16
9.2 RTSE.....	17
9.3 ACSE und ROSE.....	17
9.4 Presentation	18
9.5 ISO Session.....	18
9.6 FAX3.....	18
9.7 ISO Transport	18
9.8 Transparent Access to the ISDN B-Channel.....	19
9.9 Video Codec Controller.....	19
9.10 T.70 für ISDN	19
9.11 ISO für ISDN.....	19
10. Projekte.....	19
10.1 ITU	20
10.2 XAPI für T.120	20
10.3 T.XAPI	23
10.4 DAVIC	23
10.5 Forschungsprojekt Audio-on-Demand	24
10.6 XAPI für Windows 3.1.....	25
11. Fazit	25
12. Abkürzungen	25

XAPI

- eine universelle Kommunikationsplattform -

Von Jürgen Lohr, Berlin

Mag. Art. Jürgen Lohr,
Jahrgang 1962,
beschäftigt mit Softwareentwicklung im
Projekt "Interaktive Multimedia"
bei der Deutschen Telekom AG,
Entwicklungszentrum Berlin

1. Einleitung

Der Beitrag befaßt sich mit dem Thema der universellen Kommunikationsplattform für neue, interaktive, multimediale Dienste und Anwendungen. Ausgehend von den Diensten wird ein Referenzmodell für offene Kommunikation und die Kommunikationsplattform kurz vorgestellt. Desweiteren wird die XAPI mit den Grundbegriffen, den Phasen der Kommunikation und dem Status Modell dargelegt. Ebenfalls werden die realisierten Service Provider erläutert. Abschließend werden zukünftige Vorhaben aus den Standardisierungsprojekten ITU und DAVIC sowie weitere Realisierungen aufgezeigt.

2. Einführung

Durch den technologischen Fortschritt und den Bedarf an neuen Diensten und Anwendungen wandelt sich die klassische Telekommunikation in eine Multimediale. Sie verbindet die Nachrichtentechnik, Datenverarbeitung, Bürokommunikation, Industrieautomation und die Unterhaltungselektronik miteinander. Zu der reinen Sprach- und Datenübertragung kommen stetig steigend die Übertragung von neuen Medien (Grafik, Bilder, Videos) und neuartigen Dokumenten (gemeinsame Schreibfläche, MM-Dokumente) hinzu. Die Integration aller möglichen Medien unter einer Benutzungsoberfläche und einer Interaktivität zwischen mehreren Benutzern und dem Dienste-Anbieter-System (Service Provider) schafft eine neue Qualität der Dienste und Anwendungen.

Die Deutsche Telekom AG sieht einen großen, wachsenden Markt in diesen Multimedia-Diensten, und ihre Aufgabe liegt in der Mittlerfunktion zwischen dem Kunden und den Diensteanbietern. Da die Innovationszyklen in dieser Branche klein sind und eine Investitionssicherheit wichtig ist, wird mit starkem Engagement bei Normungsorganisationen und Standardisierungsgremien (ETSI, ITU-TS und ISO) mitgearbeitet. Hierbei wird auf offene Kommunikationssysteme und modulare Endgeräte Wert gelegt.

3. Dienstekonzept

Die neuen Multimedia-Dienste müssen drei Anforderungen erfüllen:

- Bearbeitung multimedialer Informationen (Multimedia Aspects)
- Mehrpunkt-Kommunikation (Multipoint Aspects)
- Erledigung mehrerer gleichzeitiger unterschiedlicher Aufgaben (Multitasking Aspects)

Mit Diensten und Anwendungen sind Video Konferenz, Video-on-Demand, Workflow und weitere MM-Dienste gemeint. Die Anwendungen können in folgende Bausteine (Building Blocks) unterteilt werden (Bild 1):

- Generische Teledienste (Generic Teleservices)
- Generisches Kommunikations-System
- Netzwerk Infrastruktur

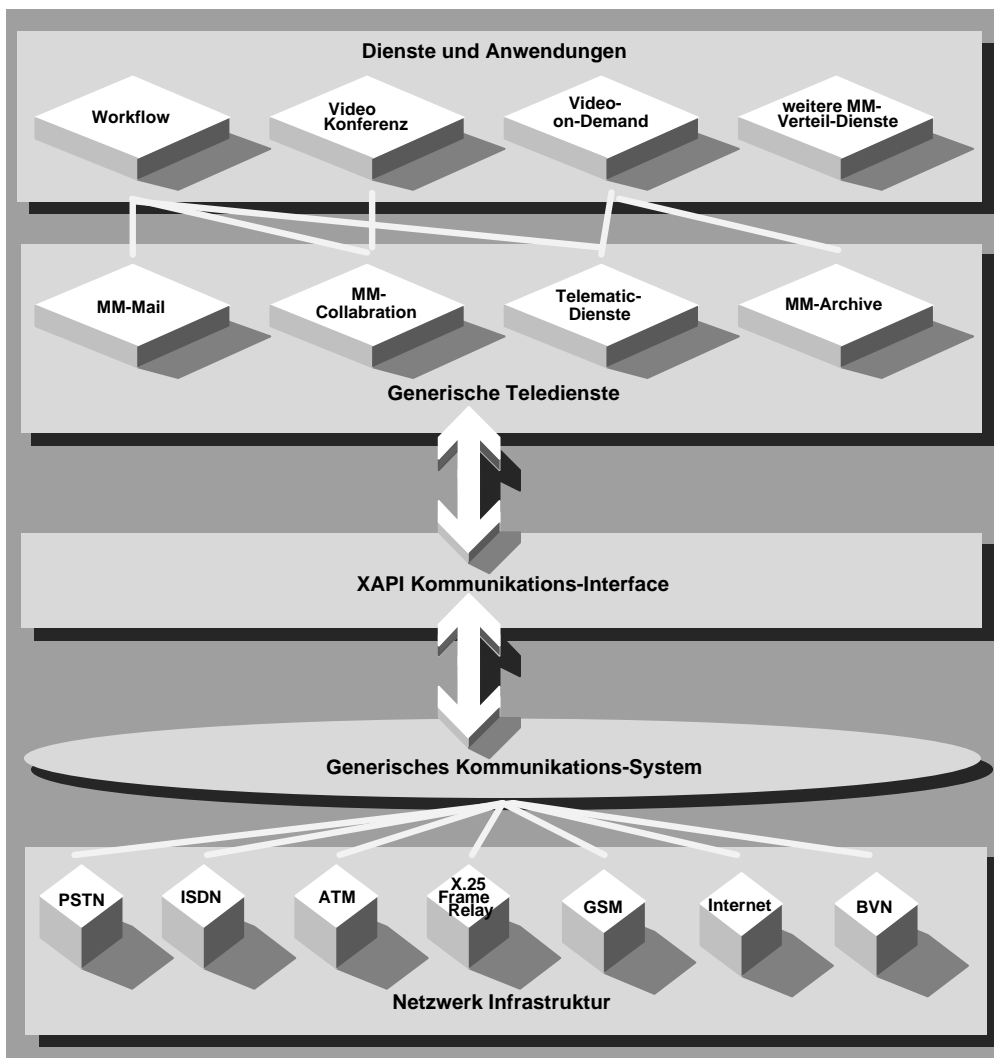


Bild 1: Architektur der Dienste und Anwendungen

Wichtig ist die Schnittstelle zwischen den Bausteinen. Das Benutzen von generischen Telediensten ist für jede Anwendung unterschiedlich. Alle Anwendungen benutzen nur einen Teil des Kommunikations-Systems. Alle Anwendungen nutzen eine oder mehrere Netzwerk-Infrastrukturen.

Eine Sichtweise einen Dienst zu beschreiben, wird mit Dienste-Kontroll-Elementen (Service Control Elements) erbracht. Die Dienste-Kontroll-Elemente sind Funktionen oder Werkzeuge, die zur Steuerung bzw. Handhabung der Dienste zur Verfügung stehen. Zwischen den Nutzern des Dienstes steht ein Diensteanbieter, der die Übertragung mit besonderen Dienstleistungen garantiert. Zwischen den Kunden und den Diensteanbietern werden Nachrichten (Service Primitives) ausgetauscht. Im Bild 2 ist das Prinzip eines Ablaufes für alle Dienste-Kontroll-Elemente dargestellt.



Bild 2: Prinzip eines Ablaufes für alle Dienste-Kontroll-Elemente

4. ISO OSI-Referenzmodell

Ein Multimedia-Kommunikationssystem ermöglicht den Austausch über Daten von unterschiedlichen Medien. Dafür benötigt man geeignete Dienste und Protokolle. Ein Dienst stellt seiner jeweiligen Anwendung eine Menge von Operationen zur Verfügung. Logisch zusammengehörige Dienste werden im Rahmen des OSI-Referenzmodells zu einzelnen Schichten zusammengefaßt. Damit erbringt eine Schicht der jeweils darüberliegenden Schicht ihren Dienst (Bild 3). Sie beschreiben das Verhalten der Schicht und ihrer Dienstelemente (Service Data Units = SDU). Die Spezifikation der Dienste enthält keine Hinweise bezüglich einer entsprechenden Implementation. Ein Protokoll besteht aus einer großen Anzahl von Referenzen, die zwischen den jeweils miteinander kommunizierenden Rechnern gelten. Weiterhin beinhaltet ein Protokoll ein Format (die Syntax) und eine Bedeutung der auszutauschenden Dateneinheiten - die Protokollateneinheiten (Protokoll Data Units = PDU). Die Partnerinstanzen auf den verschiedenen Rechnern kooperieren, um einen Dienst zu erbringen.

4.1 Bitübertragungsschicht

Die Bitübertragungsschicht definiert das Verfahren zur Übertragung einzelner Bits über das physikalische Medium. Hierbei sind das Modulationsverfahren und beispielsweise auch die Bitsynchronisation von Bedeutung. Durch die Ausbreitungsgeschwindigkeit im Übertragungsmedium, den konkret eingesetzten elektrischen Schaltkreisen, entstehen im Zusammenhang mit dem jeweiligen Modulationsverfahren Verzögerungen bei Ausbreitung der Daten. Sie bestimmen unter anderem auch die mögliche Bandbreite dieses Kommunikationskanals. Für Audio- und Videodaten muß die Verzögerung im allgemeinen minimiert werden und eine relativ hohe Bandbreite zur Verfügung stehen.

4.2 Sicherungsschicht

Die Sicherungsschicht stellt eine Übertragung einzelner Nachrichtenblöcke sicher. Im einzelnen wird das Zugriffsprotokoll (Medium Access Control = MAC) auf das physikalische Medium, Fehlererkennung und -korrektur, Flußsteuerung und Block-Synchronisation definiert. Man spricht in diesem Zusammenhang von unterschiedlichen Netzen. Kontinuierliche Datenströme erfordern die Reservierung und Gewährleistung des Durchsatzes auf einem Verbindungsabschnitt. Hier kann zur Vermeidung langer Verzögerun-

gen eine entsprechende Fehlerbehandlung erfolgen, eine Wiederholung der Übertragung ist nicht immer sinnvoll. Die geringe Fehlerrate heutiger Netze, insbesondere im Hinblick auf die Glasfasertechnologie, begünstigt ihre Verwendung für eine Übertragung multimedialer Daten. Eine feste Länge der Nachrichtenblöcke ermöglicht eine effiziente Protokollimplementierung.

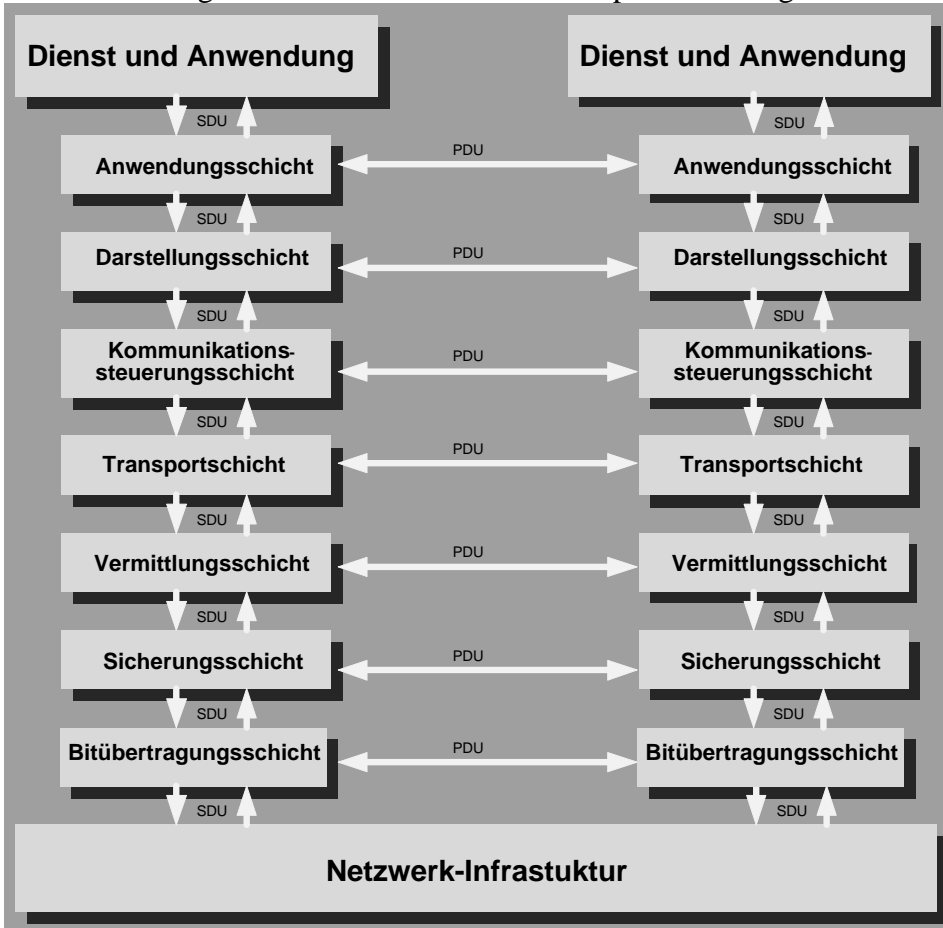


Bild 3: ISO OSI-Referenzmodell

4.3 Vermittlungsschicht

Die Vermittlungsschicht transportiert einen Nachrichtenblock (Paket) von einer Station zu einer anderen. Dies geschieht ggf. über mehrere Netze hinweg. Hier stehen Aufgaben im Bereich der Adressierung, der Vermittlung, der Wegewahl, der Fehlerbehandlung, der Netzmanagements (Congestion Control) und der Sequentialisierung der Pakete im Vordergrund. Eine Anforderung der Reservierung für die spätere Garantie der Betriebsmittel geschieht über Dienstgüteparameter, die auf kontinuierliche Datenströme eines Verbindungsabschnitts abgestimmt sind. Diese Belegung ist hier auf den gesamten Pfad zwischen den kommunizierenden Rechnern auszudehnen. Dabei ist meistens ein verbindungsorientiertes Verhalten mit dem selben Weg für alle Pakete einer Verbindung notwendig. Hiermit kann leicht die Ende-zu-Ende-Verzögerung mit einem geringen Jitter und eine korrekte Paketreihenfolge erzwungen werden. Im Bereich der Netzkopplung bzw. Netzübergänge kann für bestimmte Kommunikationsstrukturen im Multicast- und Broadcast-Verbindungen ein Duplizieren der Pakete erfolgen. Die erforderliche Dienstgüte sollte für eine Verbindung auf der Vermittlungsschicht ausgehandelt werden können.

4.4 Transportschicht

Die Transportschicht gewährleistet eine Prozeß-zu-Prozeß-Verbindung. Hierbei werden Dienstg -
temerkmale ausgehandelt und garantiert. Große Pakete werden hier segmentiert und beim Empfänger
wieder in ihre ursprüngliche Größe zusammengesetzt. Eine Fehlerbehandlung erfolgt auf Basis der Pro-
zeß-zu-Prozeß-Kommunikation. Die erforderlichen Dienstgüteparameter müssen sich auch hier auf kon-
tinuierliche Medien beziehen. Eine Fehlerbehandlung kann meistens keine Wiederholung der Datenüber-
tragung beeinhalt, weil dieses zu hohen End-zu-End-Verzögerungen und verstärktem Jitter führt. Eine
Synchronisation, die das Einhalten zeitlicher Bezüge zwischen LDUs und somit auch zwischen SDUs
verschiedener Verbindungen ermöglicht, läßt sich als Bestandteil dieser Schicht sehen.

4.5 Kommunikationssteuerungsschicht

Die Kommunikationssteuerungsschicht garantiert das Bestehen einer Verbindung während einer Sitzung.
Hier kann man zwischen unterschiedlichen Sitzungsarten in Form von Punkt-zu-Punkt, Multicast (zu
mehreren) und Multidrop (von mehreren) unterscheiden. Ein automatisiertes Wiederaufsetzen von Ver-
bindungen ist bei Sitzungen mit Kommunikation kontinuierlicher Daten nicht immer sinnvoll. Ein weite-
rer Aspekt betrifft die Kodierung der Daten. Für eine Anwendung ist es bei nachträglichem Umschalten
auf eine Verbindung von großem Nutzen zu wissen, wann eine LDU bereitsteht, die beispielsweise als
Intraframe komprimiert wurde. Nur mit einer solchen Dateneinheit kann die Präsentation beginnen.

4.6 Darstellungsschicht

Die Darstellungsschicht abstrahiert von den verschiedenen Datenformaten und bietet gemeinsame For-
mate an. Hierfür sind dann entsprechende Umwandlungsroutinen erforderlich. Ein Beispiel ist das unter-
schiedliche Format der Zahlendarstellung auf Intel- und Motorola-Prozessoren. Die Vielzahl der Audio-
und Videoformate bedingen einen Handlungsbedarf bei der Absprache und ggf. auch der Konvertierung
der Formate. Diese Problematik tritt auch außerhalb der Kommunikationskomponenten beim Austausch
von Datenträgern mit kontinuierlichen Daten auf. Deshalb wird diese Problematik oft auch außerhalb
dieser Schicht diskutiert und angesiedelt.

4.7 Anwendungsschicht

Die Anwendungsschicht betrachtet alle anwendungsspezifischen Dienste. Hierzu zählt beispielsweise die
Dateiübertragung wie das File Transfer Protocol (ftp) und die elektronische Post (e-mail). Auch hier tre-
ten spezielle Aufgaben bei der Betrachtung von Audio und Video auf. Ein entfernter Datenbankzugriff
mit gleichzeitiger Darstellung bedeutet beispielsweise hier eine Übertragung in Echtzeit mit entsprechend
vorgegebenen Parametern.

5. Kommunikationsplattform

Die diversen Multimedia-Dienste basieren auf unterschiedlichen Protokollen, Datenformaten, Netzen und
Betriebsystemen. Das Ziel einer Kommunikationsplattform für Multimedia-Dienste ist es, für diese unter-
schiedlichen Komponenten eine gemeinsame Zugangsschnittstelle für die Kommunikation zuschaffen. Die
konzeptionelle Grundlage bildet das Modell der offenen Kommunikationssysteme aus den Standardisie-
rungsempfehlungen (u.a. ITU-T T.611). Hiernach lassen sich MM-Dienste in die Bereiche der lokalen

Anwendung mit den Dienste-Funktionalitäten und der Bildschirmdarstellung sowie der kommunikativen Anwendung mit den Protokollen und den Netzzugängen unterteilen. Betrachtet man bei der Kommunikation die Dienste, die miteinander in Verbindung stehen, so müssen die Anwendungen auf einer gemeinsamen Kommunikationsplattform aufsetzen. Die Kommunikationsplattform bietet den Zugang zu den geeigneten Netzen und verwaltet den Protokollstack. Sie stellt Funktionen für die Basiskommunikation und Kontrollfunktionen für die Anwendung bereit. Die Anwendung bestimmt über Parameter die Kommunikationsdienste in den OSI-Schichten.

Die Kommunikationsplattform ist ein offenes Kommunikationsmodell, das erlaubt, daß neue Anforderungen von zukünftigen multimedialen Anwendungen mit ihren neuen standardisierten Protokollen und Netzen berücksichtigt und erweitert werden können.

Die Plattform ist ebenfalls ein integrierendes Modell, das neben dem Zugriff standardisierter OSI-Kommunikationsdienste der Ebene 7 der OSI-Referenzmodells, auch den Zugriff auf darunterliegende Ebenen des Referenzmodells gewährt. Die Plattform sieht prinzipiell die Möglichkeit vor, auf jedes Protokoll einer Ebene des Referenzmodells zuzugreifen.

Die Kommunikationsplattform bietet den Zugang zu den Netzen X.25, X21, ISDN, B-ISDN, PSTN und LAN. Das Bild 4 stellt das Gesamtmodell der integrierten Kommunikationsplattform dar. Bei dieser Darstellung ist es nicht möglich, die Beziehungen der Protokolle eindeutig zuzuordnen. Im besonderen gibt es in der Anwendungsschicht eine Überschneidung der Protokolle und der Schnittstelle. So zeigt die Darstellung nur ein Prinzip der Möglichkeiten.

Es stehen verschiedene Protokollvarianten auf der Transport- und Kommunikationssteuerungsschicht bereit, die für die Dienste genutzt werden können. Auch in der Anwendungsebene befinden sich eine große Anzahl von Protokollen, die von den Diensten angesprochen werden können. Für die Multimedia-Dienste übernehmen die Protokoll Stacks bestimmte Kommunikationsaufgaben. Grundsätzlich ist ein Protokoll Stack unabhängig von den transportierten Datentypen. Im Praktischen bestehen jedoch Abhängigkeiten durch isochrone (z.B. Audio, Video) und asynchrone Datenströme (z.B.: Text, Grafik). Daher werden bestimmte Protokoll Stacks nur für bestimmte Datentypen benutzt. Das Bild 5 zeigt den Zugriff der Dienste auf die Protokoll Stacks.

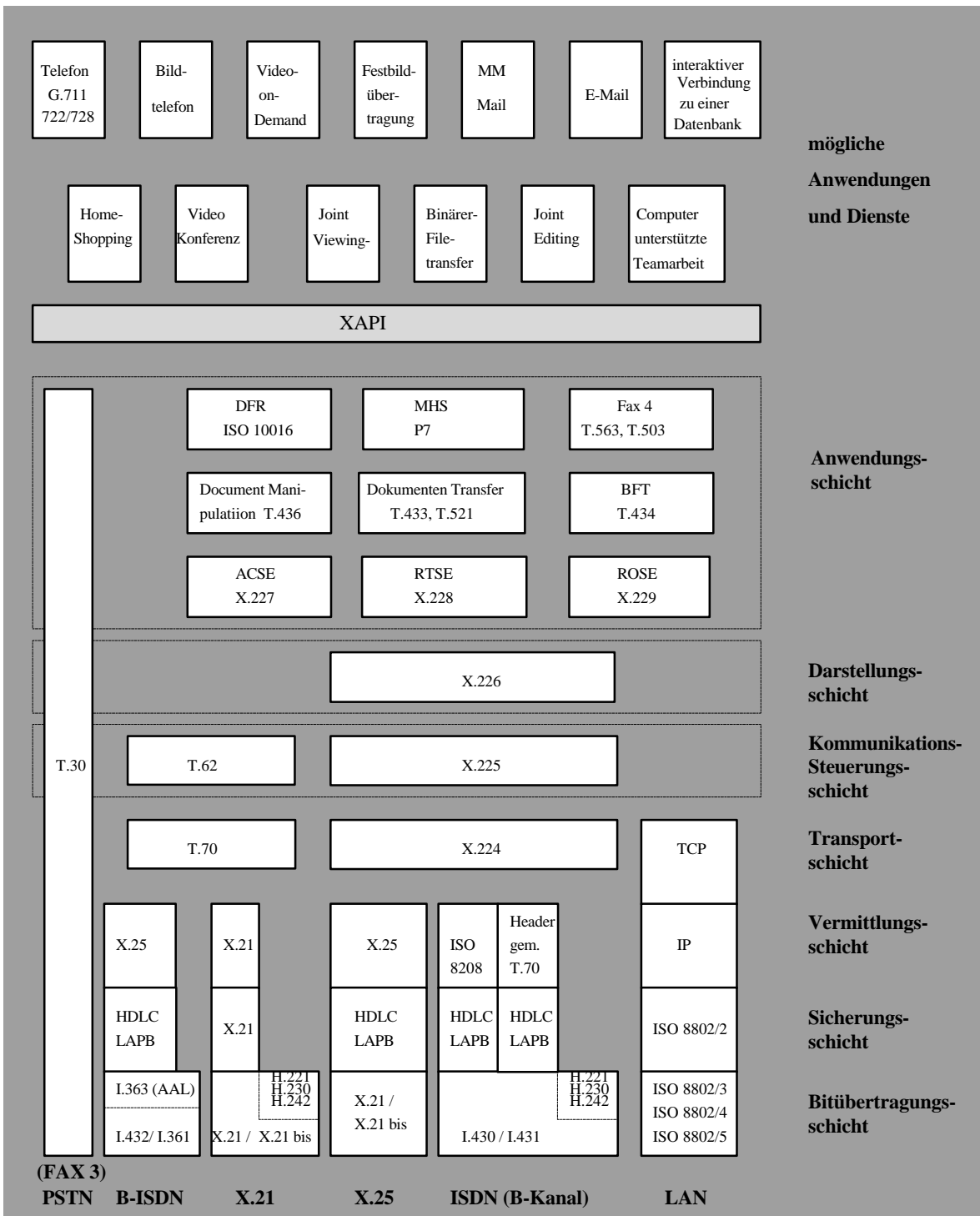


Bild 4: Das Gesamtmodell der integrierten Kommunikationsplattform

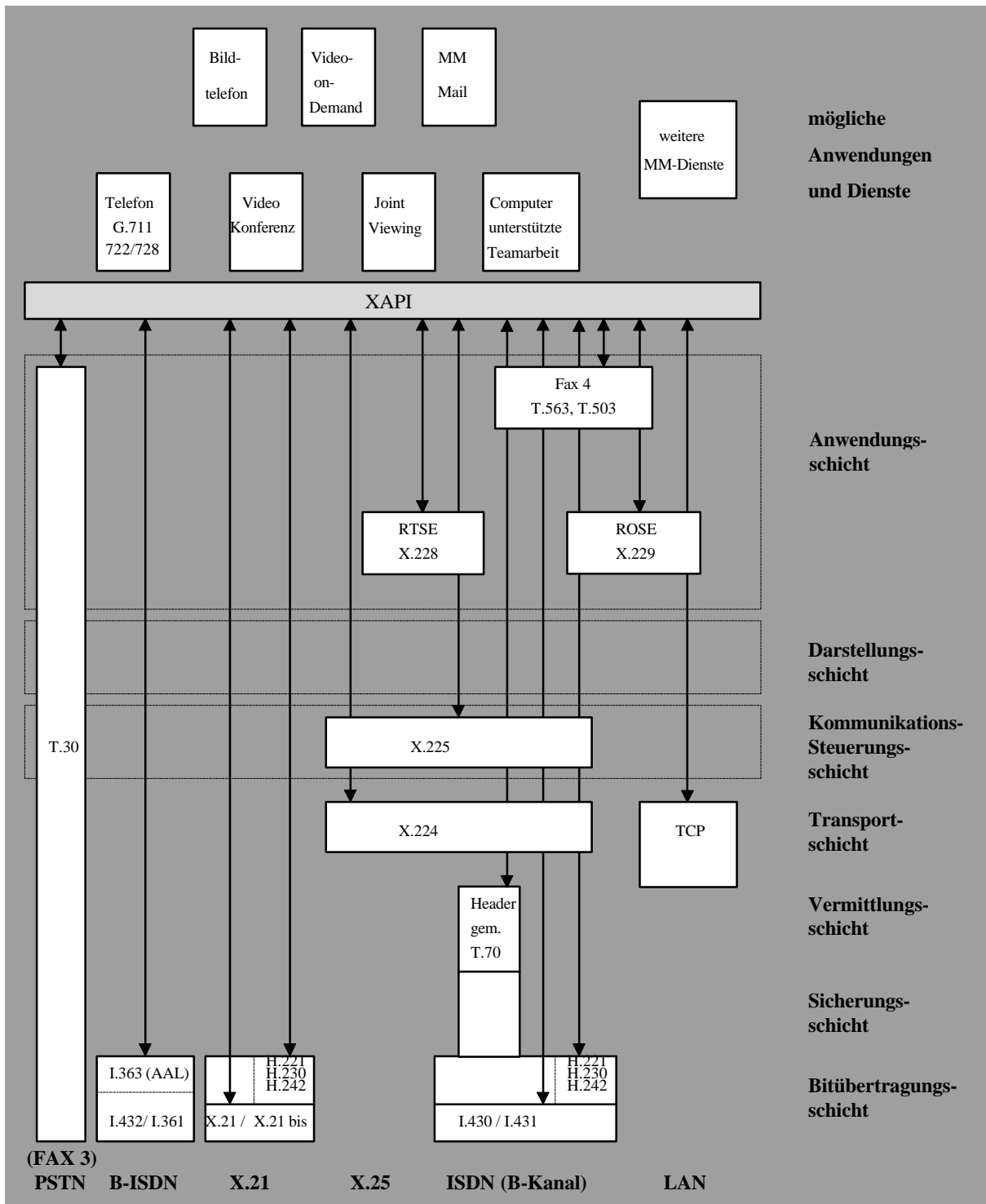


Bild 5: Zugriff der Dienste auf die Protokoll Stacks

Im folgenden werden die Kommunikationsaufgaben in den Schichten des Referenzmodells dargelegt.

5.1 Zugriff auf die Bitübertragungsschicht

Auf der physikalischen Ebene der Bitübertragungsschicht ist ein Zugriff nur für isochrone Datenströme sinnvoll. Diese sind für die kontinuierlichen Medien wie Audio und Video gedacht, die in Echtzeitanwendungen eingesetzt werden. Durch den direkten Zugriff auf die unterste Schicht im Referenzmodell entstehen starke Abhängigkeiten vom MM-Dienst zum eingesetzten Netz. Das widerspricht der Philosophie

der universellen, einsetzbaren Kommunikationsplattform. Durch den direkten Zugriff entfallen ebenfalls Sicherungsmechanismen für die Daten. Daher sollten die isochronen Daten möglichst über sichere Netzen, also digitale Netzen wie X.21, ISDN und B-ISDN, übertragen werden.

5.2 Zugriff auf die Transportschicht

Auf die Transportschicht ist ein Zugriff von einer Anwendung mit asynchronen Datenströmen geeignet. Die netzunabhängigen Datenströme können u.a. Text oder Binärdaten enthalten, die gesichert übertragen werden sollen.

5.3 Zugriff auf die Kommunikationssteuerungsschicht

Auf die Kommunikationssteuerungsschicht ist ein Zugriff für Interaktive Dienste geeignet, die zu den Schichtdiensten der Transportschicht auch Daten zur Dialogsteuerung benötigen. Diese Datenströme können Informationen von Tastatur und Maus auf gemeinsame Dokumente sein. Bei den Diensten wie Joint Editing werden neben den Dialogsteuerungsdaten auch die Sendeberechtigungen koordiniert.

5.4 Zugriff auf die Anwendungsschicht

Innerhalb der Anwendungsschicht existieren unterschiedlich hierarchische Protokollstacks mit ihren unterschiedlichen Diensten. Unter den Basisdiensten sind ACSE (Association Control Service Element), RTSE (Reliable Transfer Service Element) und ROSE (Remote Operation Service Element) zu verstehen. ACSE führt bei interaktiven Diensten den Verbindungsaufbau und -abbau durch. ROSE ermöglicht den interaktiven Diensten komfortabel auf entfernte Dokumentendatenbanken wie Festbilder oder MM-Dokumente zuzugreifen. RTSE gestattet, große Datenmengen beliebigen Informationstyps zwischen den Kommunikationspartnern zu übertragen. Der Binäre Dateitransferdienst BFT (Binary File Transfer) ermöglicht, beliebiggeartete Binärdateien zwischen zwei Anwendungen zu versenden. Der FAX4-Dienst verschickt Datenströme vom Typ der Rastergrafik, wobei eine Kodierung nach T.4 oder T.6 vorgeschrieben ist.

6. XAPI

Die XAPI ist die Kurzform von dem Begriff "eXtensive Application Programming Interface" - eine erweiterte Schnittstelle für die Programmierung von Anwendungen und Diensten. Sie basiert auf dem Modell von "X/Open Transport Interface (XTI)" und dem "UNIX System V Release 3 Transport Level Interface (TLI)". Es wurden die prinzipiellen Mechanismen der XTI übernommen. So wurden beim XAPI Service Provider die spezifischen Argumente in den Speichern der einzelnen Schichten übernommen und auf alle Schichten des Referenzmodells erweitert. In einigen Punkten geht die XAPI dem Konzept der XTI weiter. So könnten die unteren Schichten, im besonderen die physikalische Schicht, über Service Primitive angesprochen werden. Die XAPI ist eine konkrete Implementation und besitzt nicht den Anspruch einer allumfassenden Schnittstelle wie XTI. Sie ist jedoch leicht erweiterbar auf neue Belange.

Die XAPI bietet eine Kommunikationsplattform für übergreifende Kommunikationsdienste, die eine Programmierschnittstelle in der Sprache "C" für die Anwendungen und Dienste bereitstellt. Sie ist nicht be-

stimmt für eine bestimmte Schicht, sondern erlaubt einen Zugriff auf alle Schichten des OSI-Referenzmodells. Sie bietet einen Rahmen für Dienste unabhängiger Bibliotheksfunktionen, die den Diensten für Aufgaben der Kommunikation zur Verfügung stehen. Hier gibt es Basisfunktionen, um einen Austausch von Dienste Primitiven zwischen dem Nutzer-Prozeß und dem XAPI Service Provider zu ermöglichen. Weiterhin existieren komplexere Funktionen, die eine verbindungsorientierte Kommunikation

Der installierte XAPI Service Provider ermöglicht Kommunikationsdienste. Die XAPI ermöglicht nur einen Zugangsmechanismus zu diesen. Neue XAPI Service Provider können einfach zur XAPI hinzugefügt werden. Genauso einfach kann zu einem bestehenden XAPI Service Provider ein einzelnes Protokoll ausgetauscht werden. Diese Kommunikationsplattform soll für die speziellen Anforderungen und alle Kommunikationseigenschaften eine homogene Schnittstelle bieten.

6.1 Dienste Endpunkt

Der Dienste Endpunkt (Service Endpoint) spezifiziert den Kommunikationsweg zwischen einem Dienste Nutzer und einem speziellen XAPI Service Provider. Der Dienste Endpunkt wird für die Funktion `x_open` angelegt, welches einen lokalen Bezeichner (file descriptor) zurückliefert. Dieser File Descriptor dient bei allen Funktionen der XAPI als Identifikator des Dienste Endpunktes. Ein Dienst Endpunkt kann nur eine Verbindung zu einer Zeit verwalten. Aber eine Anwendung kann mehrere Dienste Endpunkte anlegen, wenn es notwendig wird. Nach dem Anlegen eines Dienste Endpunktes muß dieser aktiviert werden, welches mit der Funktion `x_bind` geschieht. Dann erst kann eine Kommunikation erfolgen. Bei dem Aktivieren des Dienste Endpunktes wird ein Bezug von einer Adresse zu einem Endpunkt hergestellt und eine Verbindung des Anwendungssystems mit einem Transportsystem herbeigeführt. Jetzt besteht ein aktiver Dienste Endpunkt (aktive service endpoint).

6.2 Protokoll Module

Ein Protokoll Modul ist eine Implementation eines Kommunikationsprotokolls, welches einen XAPI-Dienst bereitstellt. Es ist für eine Schicht des OSI-Referenzmodells vorgesehen. Jedes Modul besitzt eine Identifizierungsnummer, den Protokoll Modul Bezeichner (protocol module identifier).

6.3 Protokoll Stack

Ein Protokoll Stack wird gebildet aus zwei bis mehreren Protokoll Modulen. Jedes Modul in dem Stack benutzt das tieferliegende Protokoll mit seinen Schichten-Diensten. Das unterste Protokoll in der untersten Schicht - der Bitübertragung, benutzt den Hardware Device Treiber. In einem Protokoll Stack bietet das oberste Protokoll seine Dienste der Anwendung an. Dabei werden die untersten Protokolle für die Anwendung versteckt.

In der XAPI Terminologie existieren zwei Klassen von Protokoll Stacks. Bei einem Applikationssystem werden Zugriffe auf alle oder einige Protokolle der Kommunikationssteuerungs-, der Präsentations- und der Anwendungsschicht gewährt. Bei einem Transportsystem werden die Zugriffe auf alle oder einige Protokolle der Bitübertragungs-, der Sicherungs-, der Vermittlungs- und der Transportschicht gewährt.

6.4 Service Primitive

Die Service Primitive beschreiben ein spezielles Protokoll einer Schicht aus einer Ansammlung von abhängigen Protokollen - dem Protokoll Stack in der XAPI Bibliothek. Die Service Primitive Parameter konvertieren Informationen zwischen der Anwendung und dem Service Provider. Die Informationen sind für die Gegenstelle bestimmt - dem Remote User. Daher werden sie manchmal in Protokoll Data Units (PDU) umgewandelt.

6.5 Service Provider

Ein Service Provider bietet mit einem Protokoll Stack einen Kommunikationsdienst einer Anwendung über eine Schnittstelle an. Ein Transport Service Provider bietet als obersten Dienst den Transportdienst im OSI-Referenzmodell an. Jeder Provider besitzt einen eindeutigen Bezeichner (service provider identifier), der bei der Funktion `x_open` als Parameter angegeben werden muß, um den passenden Service Provider zu aktivieren.

Unterschiedliche Service Provider befinden sich auf der gleichen Schicht im Referenzmodell. Sie können unterschiedliche Protokoll Stacks bieten. Ebenso teilen sich unterschiedliche Service Provider tieferliegende Protokolle.

6.6 Prozeß Endpunkt

Eine Anwendung kann gleichzeitig bis zu mehreren Service Providern öffnen. Jeder neue Endpunkt wird mit der Funktion `x_open` angelegt und mit einem eindeutigen Bezeichner - dem File Descriptor - versehen. Jeder Endpunkt ist vom Anderen unabhängig und kann daher auch gleichzeitig mit unterschiedlichen Protokollen aktiviert werden.

6.7 Synchroner und Asynchroner Mode

Die XAPI bietet die beiden Modi Synchron und Asynchron an, in dem die XAPI arbeiten kann (execution mode). Im Synchronen Mode arbeitet die XAPI-Bibliotheksfunktionen so ab, daß beim Absetzen von Ereignissen am Dienst Endpunkt auf ein ankommendes Ereignis gewartet wird. Erst wenn das Ereignis eingetroffen ist, erhält der Nutzer die Kontrolle wieder. Daher blockiert der Nutzerprozeß das ganze System und kein Hintergrunds- bzw. zweiter Prozeß kann ausgeführt werden. So werden gegebenenfalls auch die weiteren geöffneten Service Endpoints unterbrochen. Daher ist dieser Mode nur für eine Verbindung geeignet.

Der Asynchrone Mode übergibt ein Ereignis des Service Endpoints der XAPI und erhält die Kontrolle des Systems zurück. Die XAPI wartet hier nicht auf eintreffende Ereignisse. So kann die Anwendung einige Aufgaben weiterhin erfüllen. Die Anwendung muß jedoch periodisch auf eintreffende Ereignisse die XAPI abfragen. Daher ist dieser Mode für Anwendungen geeignet, die eine lange Reaktion des Kommunikationspartners und weitere Aufgaben neben der Kommunikation zulassen. So arbeiten Anwendungen in diesem Mode, die z.B. interaktive Dialoge besitzen oder mehrere Partner miteinander verbinden.

7. Phasen der Datenkommunikation

Die Phasen der Datenkommunikation in einem verbindungsorientierten Dienst (Connection- Mode Service - COS) sind die Initialisierung, der Verbindungsaufbau, der Datentransfer, der Verbindungsabbau und die Deinitialisierung.

In der Initialisierungsphase werden die Ressourcen verwaltet und die passive bzw. die aktive Rolle des Kommunikationspartners festgelegt. Mit der Funktion `x_open` wird ein Service Endpunkt für den Service Provider angelegt. Mit der Funktion `x_bind` wird der Service Endpunkt für eine bestimmte Adresse aktiviert.

In der Verbindungsaufbauphase werden die Dienste Parameter zwischen den Partnern ausgetauscht und eine Verbindung hergestellt. Dabei stehen den aktiven und passiven Seiten andere Funktionen zur Verfügung. Auf der aktiven Seite kann mit der Funktion `x_conreq` ein Verbindungsaufbau mit einer Partneradresse, den Dienste Parametern und Nutzer Daten eingeleitet werden. Weiterhin kann man mit der Funktion `x_conconf` überprüfen, ob die Verbindung mit der Partneradresse, den Dienste Parametern und den Nutzer Daten zustandegekommen ist. Auf der passiven Seite kann mit der Funktion `x_conind` ein Verbindungswunsch mit einer Anruferadresse, den Dienste Parametern und Nutzer Daten angezeigt werden. Weiterhin kann mit der Funktion `x_conrsp` dem Verbindungswunsch mit der eigenen Adresse, den Dienste Parametern und den Nutzer Daten abgelehnt oder zugestimmt werden.

Bei den Partnern stehen in der Datentransferphase die Funktionen `x_snddata`, `x_rvcdata`, `x_sndsp`, und `x_rvcsp` zur Verfügung. Die Funktion `x_snddata` ermöglicht Daten der beiden Kommunikationspartner mit Dienste Parameter und Nutzer Daten abzusenden. Die Funktion `x_rvcdata` ermöglicht die Daten der beiden Kommunikationspartner mit Dienste Parameter und Nutzer Daten. Die Funktion `x_sndsp` ermöglicht den Dienste-Primitiven (service primitives), den lokalen Provider mit Dienste Parameter und Nutzer Daten zu übergeben. Die Funktion `x_rvcsp` ermöglicht, die Dienste Primitiven dem lokalen Provider mit Dienste Parameter und Nutzer Daten zu entnehmen.

In der Verbindungsabbauphase wird das Ende einer Kommunikation zwischen den Partnern eingeleitet und eine Verbindung geschlossen. Bei einer ordentlichen Auflösung der Verbindung (orderly release) stehen der aktiven und passiven Seite andere Funktionen zur Verfügung, als bei einer abrupten Auflösung der Verbindung (abortive release). Auf der aktiven Seite kann bei einem ordentlichen Abschluß mit der Funktion `x_relreq` ein Verbindungsabbau mit den Dienste Parametern und Nutzer Daten eingeleitet werden. Weiterhin kann mit der Funktion `x_relconf` angezeigt werden, ob die Verbindung mit den Dienste Parametern und den Nutzer Daten abgebaut worden ist. Mit der Funktion `x_rcvend` kann nun angezeigt werden, daß der Endpunkt wieder freigegeben wurde. Auf der passiven Seite kann mit der Funktion `x_relind` ein Verbindungsabbauwunsch des Partners mit den Dienste Parametern und Nutzer Daten angezeigt werden. Weiterhin kann mit der Funktion `x_relrsp` der Wunsch der Verbindungsauflösung mit den Dienste Parametern und den Nutzer Daten bestätigt werden. Auf der passiven Seite kann mit der Funkti-

on `x_rcvend` angezeigt werden, ob der Endpunkt freigegeben wurde. Im Fall einer abrupten Auflösung stehen den beiden Partnern die Funktionen `x_snddis` - zum Abbruch einer Verbindung, `x_rcvdis` - zur Anzeige einer abgebrochenen Verbindung und `x_rcvend` - zum Anzeigen eines freigegebenen Endpunktes zur Verfügung. Bei einer ordentlichen Auflösung einer Verbindung wird garantiert, daß die Daten, die sich auf dem Weg zwischen den Partnern befinden, vollständig ausgetauscht werden.

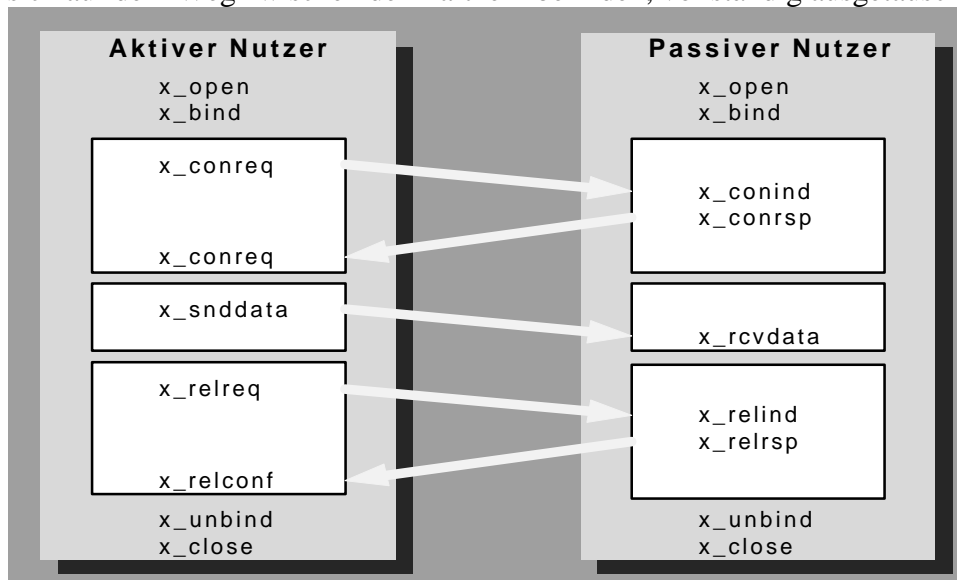


Bild 6: Sequenz von Funktionsaufrufen

In der Deinitialisierungsphase werden wie in der Initialisierungsphase die Ressourcen verwaltet. Mit der Funktion `x_unbind` wird der Service-Endpunkt für eine bestimmte Adresse deaktiviert. Mit der Funktion `x_close` wird ein Service-Endpunkt für den Service Provider gelöscht.

Die meisten Funktionen werden von der Anwendung mit User Daten und Service Parametern übergeben. In einigen Fällen können diese Daten in Fragmente aufgeteilt werden, die in den Funktionen mit dem Kennzeichen "X_Moreflage" aufgezeigt werden. In Bild 6 ist eine einfache Sequenz von Funktionsaufrufen für den aktiven und den passiven Nutzer dargestellt.

8. Status Modell

Eine genauere Beschreibung der Abfolgen der XAPI Funktionen kann in einem Status Modell (state model) dargelegt werden (Bild 7). Dabei werden die Status Zustände in Objekten mit XAPI-Sate und die Aktionen als Pfeile mit XAPI-Funktionen dargestellt. In diesen Diagrammen wird zum einen der aktive und zum anderen der passive Verbindungsaufbau und -abbau dargestellt. Dabei wird nicht berücksichtigt, daß die Nutzer Daten, die in den Funktionen übergeben werden, nicht in Fragmente zerteilt werden. Ebenfalls soll hier verdeutlicht werden, daß nicht nur der aktive Nutzer eine Beendigung der Kommunikation einleiten kann.

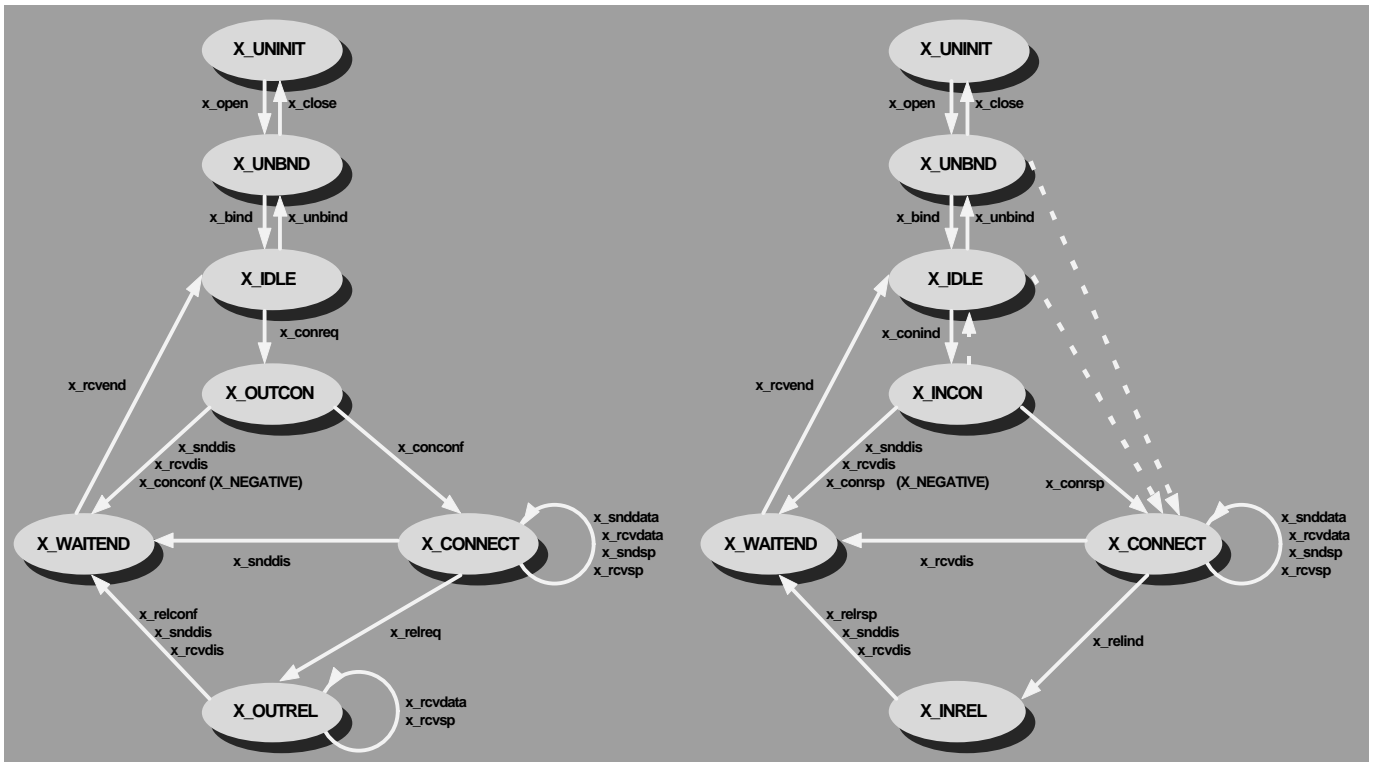


Bild 7: Status Modell

Folgende XAPI-States existieren:

- X_UNINIT Der Dienst Endpunkt ist noch nicht angelegt.
- X_UNBIND Der Dienst Endpunkt ist noch nicht aktiviert, keine Adresse ist be-
kannt und kein Transport-System ist ausgewählt.
- X_IDLE Der Dienst Endpunkt ist aktiviert. Jedoch sind noch keine Kommu-
nikations-, Übertragungs-Qualitäten und -Quantitäten ausgewählt.
- X_OUTCON Der Endpunkt ist aktiv. Ein Verbindungswunsch ist abgesetzt. Es
wird auf eine Bestätigung des Partners Service Provider gewartet.
- X_INCON Der Endpunkt ist aktiv. Ein Verbindungswunsch ist abgesetzt. Es
wird auf eine Bestätigung des Partners gewartet.
- X_CONNRCT Eine Verbindung zum Partner besteht. Man befindet sich in der Da-
tenübertragungsphase.
- X_INREL Eine ordentliche Auflösung der Kommunikation wurde eingeleitet.
Auf eine Bestätigung vom Partner wird gewartet.
- X_OUTREL Eine ordentliche Auflösung der Kommunikation wurde eingeleitet.
Auf eine Bestätigung des Partners Service Provider wird gewartet.
- X_WAITEND Die bestehende Verbindung hat den Abbauwunsch erhalten. Nun
wartet der Endpunkt auf ein Zeichen des Service Providers, daß die
Verbindung geschlossen wurde und eine neue Verbindung aufgebaut
werden kann.

9. Service Provider

Mittlerweile sind für mehrere XAPI Service Provider und XAPI Transportsysteme verfügbar. Ein Service Provider bietet seine Dienste der Kommunikations-Protokoll Stack dem Nutzer über eine hohe Schnittstelle an. Jeder XAPI Service Provider wird über einen eindeutigen Namen identifiziert. Dieser Name besteht aus Buchstaben (Typ: string) und wird als Parameter bei der Funktion `x_open` mitgegeben, um den XAPI Service Provider anzuwählen.

Die XAPI Service Provider können in die Arten unterteilt werden:

- Applikationssystem (einige oder alle der Schichten 5,6 und 7),
- Transportsystem (einige oder alle der Schichten 1 bis 4) und
- Applikationssystem mit Transportsystem (einige oder alle Schichten 1 bis 7).

Um einen XAPI Service Provider eines Applikationssystems für die Kommunikation benutzen zu können, muß ein Transportsystem angebunden werden. Dieses wird dynamisch bei dem Funktionsaufruf `x_bind` erreicht, indem der Name des Transportsystems als Parameter der Funktion beigegeben wird. Die anderen Arten der XAPI Service Provider benötigen kein dynamisches Binden des Transportsystems. Aber auf jeden Fall wird der Service Endpunkt mit diesem Provider verbinden, wenn sie mit der Funktion `x_bind` aktiviert wird.

Die folgenden XAPI Service Provider sind zur Zeit verfügbar:

- FAX4 und BFT, Version 1.0, 1995
- RTSE, Version 1.0, 1995
- ACSE, Version 1.0, 1995
- ROSE, Version 1.0, 1995
- Presentation, Version 1.0, 1995
- ISO Session, Version 1.0, 1995
- FAX3, Version 1.0, 1995
- ISO Transport, Version 1.0, 1995
- Transparent Access to the ISDN B-Channel, Version 1.0, 1995
- Video Codec Controller, Version 1.0, 1995

Die folgenden XAPI-Transportsysteme sind zur Zeit verfügbar:

- T.70 für ISDN, Version 1.0, 1995
- ISO für ISDN, Version 1.0, 1995

9.1 FAX4 und BFT

Der Bezeichner "`X_FAX4_BFT_ISDN`" aktiviert den Service Provider für die Dienste FAX 4 und Binary File Transfer über ISDN. Der Protokoll Stack bietet folgende Dienste auf den Schichten des OSI-Referenzmodells:

- Schicht 7: T.433 DTAM transparenter Mode

T.521 DAP für Bulk Transfer

T.503 DAP Austausch für Faksimile der Gruppe 4

T.563 Terminal Merkmale für Faksimile der Gruppe 4

- Schicht 6: T.62
- Schicht 5: T.62
- Schicht 4: T.70
- Schicht 3: T.90
- Schicht 2: X.75 LAPB
- Schicht 1: I.430
I.431

9.2 RTSE

Der Bezeichner "X_RTSE_ISDN" aktiviert den Service Provider für den Dienst Reliable Transfer Service Element über ISDN. Der Protokoll Stack bietet folgende Dienste auf den Schichten des OSI-Referenzmodells:

- Schicht 7: X.228 RTSE Normaler Mode
X.227 ACSE
- Schicht 6: X.226
- Schicht 5: X.225
- Schicht 4: X.224
- Schicht 3: ISO 8208
- Schicht 2: X.75 LAPB
- Schicht 1: I.430
I.431

9.3 ACSE und ROSE

Der Bezeichner "X_ACSE_ROSE_ISDN" aktiviert die Service Provider für die Dienste Association Control Service Element und Remote Operation Service Element über ISDN. Der Protokoll Stack bietet folgende Dienste auf den Schichten des OSI-Referenzmodells:

- Schicht 7: X.227 ACSE
X.229 ROSE
- Schicht 6: X.226
- Schicht 5: X.225
- Schicht 4: X.224
- Schicht 3: ISO 8208
- Schicht 2: X.75 LAPB
- Schicht 1: I.430
I.431

9.4 Presentation

Der Bezeichner "X_ACSE_PRES_ISDN" aktiviert den Service Provider für den Dienste Association Control Service Element auf der Präsentationsschicht über ISDN. Der Protokoll Stack bietet folgende Dienste auf den Schichten des OSI-Referenzmodells:

- Schicht 7: X.227 ACSE
- Schicht 6: X.226
- Schicht 5: X.225
- Schicht 4: X.224
- Schicht 3: ISO 8208
- Schicht 2: X.75 LAPB
- Schicht 1: I.430
I.431

9.5 ISO Session

Der Bezeichner "X_ISO_SESSION_ISDN" aktiviert den Service Provider für den Dienst ISO auf der Kommunikationssteuerungsschicht über ISDN. Der Protokoll Stack bietet folgende Dienste auf den Schichten des OSI-Referenzmodells:

- Schicht 5: X.225
- Schicht 4: X.224
- Schicht 3: ISO 8208
- Schicht 2: X.75 LAPB
- Schicht 1: I.430
I.431

9.6 FAX3

Der Bezeichner "X_FAX3_ISDN" aktiviert den Service Provider für den Dienst Faksimile der Gruppe 3 über ISDN. Der Protokoll Stack bietet den folgenden Dienst auf der Schicht des OSI-Referenzmodells:

- Schicht 1: T.30 Adaptionsschicht

9.7 ISO Transport

Der Bezeichner "X_Transport_ISDN" aktiviert den Service Provider für den Dienst ISO auf der Transportschicht über ISDN. Der Protokoll Stack bietet folgende Dienste auf den Schichten des OSI-Referenzmodells:

- Schicht 4: X.224
- Schicht 3: ISO 8208
- Schicht 2: X.75 LAPB
- Schicht 1: I.430
I.431

9.8 Transparent Access to the ISDN B-Channel

Der Bezeichner "X_PHT_ISDN" aktiviert den Service Provider für den Dienst des transparenten Zugangs auf die physikalische Schicht über ISDN. Der Protokoll Stack bietet folgende Dienste auf den Schichten des OSI-Referenzmodells:

- Schicht 1: I.430
I.431

9.9 Video Codec Controller

Der Bezeichner "X_PHC_ISDN" aktiviert den Service Provider für den Dienst, direkt einen Audio und Video-Codec auf der physikalischen Schicht über ISDN im B-Kanal zu kontrollieren. Der Protokoll Stack bietet den folgenden Dienst auf der Schicht des OSI-Referenzmodells:

- Schicht 1: Adaptive Schicht zu dem A/V-Codec

9.10 T.70 für ISDN

Der Bezeichner "X_TRSYS_T70_ISDN" aktiviert ein Transportsystem für den Dienst T.70 über ISDN im B-Kanal und bietet einen Protokoll Stack mit den folgenden Diensten auf den Schichten des OSI-Referenzmodells:

- Schicht 4: T.70
- Schicht 3: T.90
- Schicht 2: X.75 LAPB
- Schicht 1: I.430
I.431

9.11 ISO für ISDN

Der Bezeichner "X_TRSYS_ISO_ISDN" aktiviert ein Transportsystem für die Dienste ISO 8208 über ISDN im B-Kanal. Der Protokoll Stack bietet folgende Dienste auf den Schichten des OSI-Referenzmodells:

- Schicht 4: X.224
- Schicht 3: ISO 8208
- Schicht 2: X.75 LAPB
- Schicht 1: I.430
I.431

10. Projekte

Um eine Investitionssicherheit für interaktive Multimediale Dienste und Anwendungen zu erreichen, bemüht sich die Deutsche Telekom AG, ihre Vorstellungen und Interessen in Standardisierungsorganisationen einzubringen. Dies erfolgt im starken Maße mit der XAPI, die in verschiedenen Vorhaben von Standardisierungsprojekten der ITU und DAVIC als auch durch Realisierung von Projekten für Audio-on-Demand und Videokonferenz eingebracht wird.

10.1 ITU

Die internationale Telecommunication Union (ITU) besteht aus einem Zusammenschluß von vielen bedeutenden Telekommunikationsgesellschaften aus der gesamten Welt. Hier werden Übereinkünfte über weltweit geltende Standardisierungen beschlossen. Das Verfahren, ein Thema zu standardisieren, hat ein festgelegtes Vorgehen. So gibt es zu speziellen Themen Studienkommissionen (study groups), die sich mit bestimmten Teilthemen - den Studienfragen - beschäftigen. Neu ist der Ansatz, nicht mehr Architekturen und Plattformen in den Vordergrund zu stellen, sondern marktorientierte Systemlösungen in Form von Standardisierungsprojekten der ITU-T durchzuführen.

Die Studiengruppe 8 der ITU, die sich mit neuen Multimedia-Anwendungen beschäftigt, hat die Wichtigkeit einer Kommunikationsplattform der XAPI erkannt und sich zum Ziel gesetzt, sie in die neuen MM-Diensten zu festigen. Weiterhin soll in einigen Studienfragen eine Überschneidung zu diesem Thema überprüft werden. So wird stark an der Realisierung der Empfehlung der T.XAPI gearbeitet.

Der T.120 Standard bietet allen Kommunikationsanwendungen einen Mehrpunkt-Verbindungsdienst, der über mehrere Komponenten erbracht wird (Bild 8). Die Komponente Node Controller (NC) bietet Management Funktionen an. Die Komponente Application Protocol Entities (APE) stellt ein Protokoll für eine Anwendung bereit. Die Komponente Generic Conference Control (GCC) bietet einen Satz von Diensten für das Konfigurieren und das Management von Mehrpunkt Konferenzen an. Die Komponente Multipoint Communication Service (MCS) bietet einen globalen Datendienst für Mehrpunkt Kommunikation an. Es verwaltet Punkt-zu-Punkt-Verbindungen und setzt die Informationsvermittlung für Mehrpunkt-Verbindungen um. Ein große Anzahl von logischen Kanälen kann die Datenübertragung mit Direkt-Verbindung (Eins zu Eins), Broadcast-Verbindung (Eins zu Viele) und Sammel-Verbindung (Viele zu Eins) verwalten.

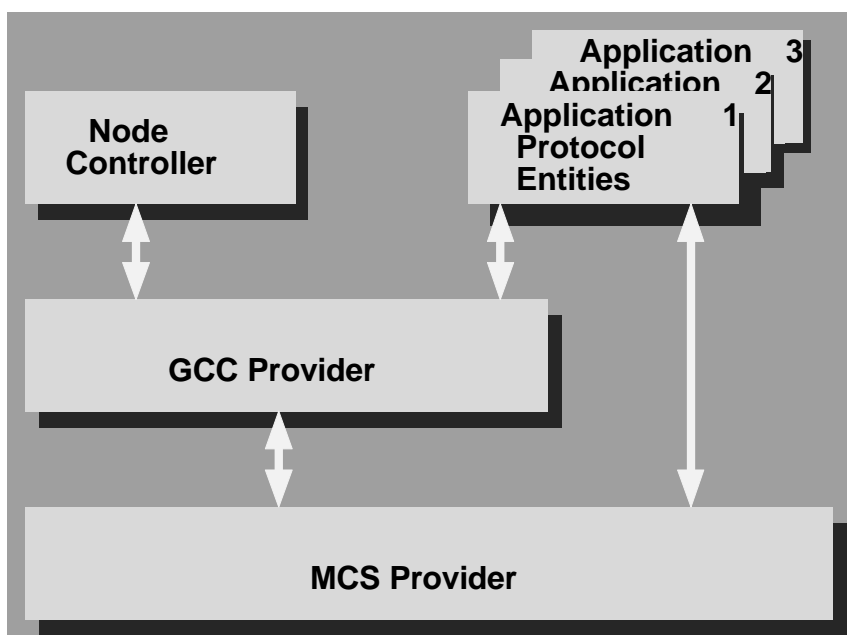


Bild 8: System Modell des Standards T.120

Die XAPI bietet für T.120 folgende Vorteile:

- Die Kommunikationsbelange werden dem Nutzer ausgeblendet.
- Sie ist abbildbar in einem weiteren Bereich von Multimedia und Multipoint Anwendungen und Diensten.
- Sie unterstützt eine große Anzahl von Netzinfrastrukturen
- Sie bietet dem Nutzer einen flexiblen und einheitlichen Zugang zu Kommunikationsdiensten
- Sie ermöglicht hohe Performance der Informationsübertragung über die Schnittstelle.

Im Konferenzumfeld kann unterschieden werden zwischen dem Konferenz Nutzer und dem Konferenz Provider (Bild 9). Der Konferenz Nutzer kann einen Zugang zu Kontroll- und Anwendungsfunktionen haben. Der Konferenz Provider bietet Mehrpunkt (Multipoint) Kommunikation an, bei dem die Übertragung der Information an eine größere Anzahl von Anwendungen zur gleichen Zeit erfolgt. Weiterhin bietet der Provider eine Multimedia-Kontrolle, bei dem er die Realzeit-Datenströme von Audio und Video einer Quelle zu mehreren Zielen überträgt. Auch bietet er ein Protokoll für die Anwendungen und Dienste. Der Konferenz Provider bietet auch während der Konferenz eine Kontrolle, so z.B. die Änderung der Verbindungsmerkmale und die Koordination der Sende- und Empfangszuordnung.

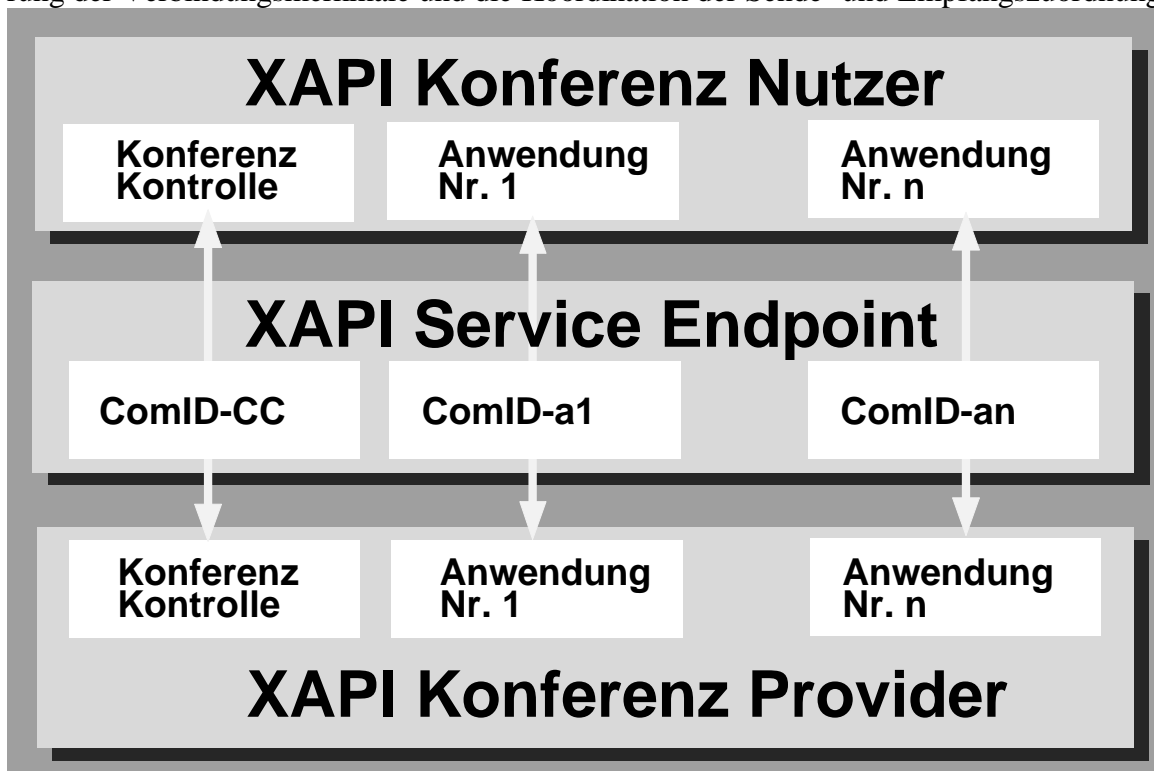


Bild 9: Das generische Modell einer Konferenz

In Bild 10 ist der Zugriff der Dienste auf die Protokoll Stacks für T.120 dargestellt. Der Remote Call (MRPC) und der Multipoint Binary File Transfer (MBFT) sind Module aus dem Standard T.120.

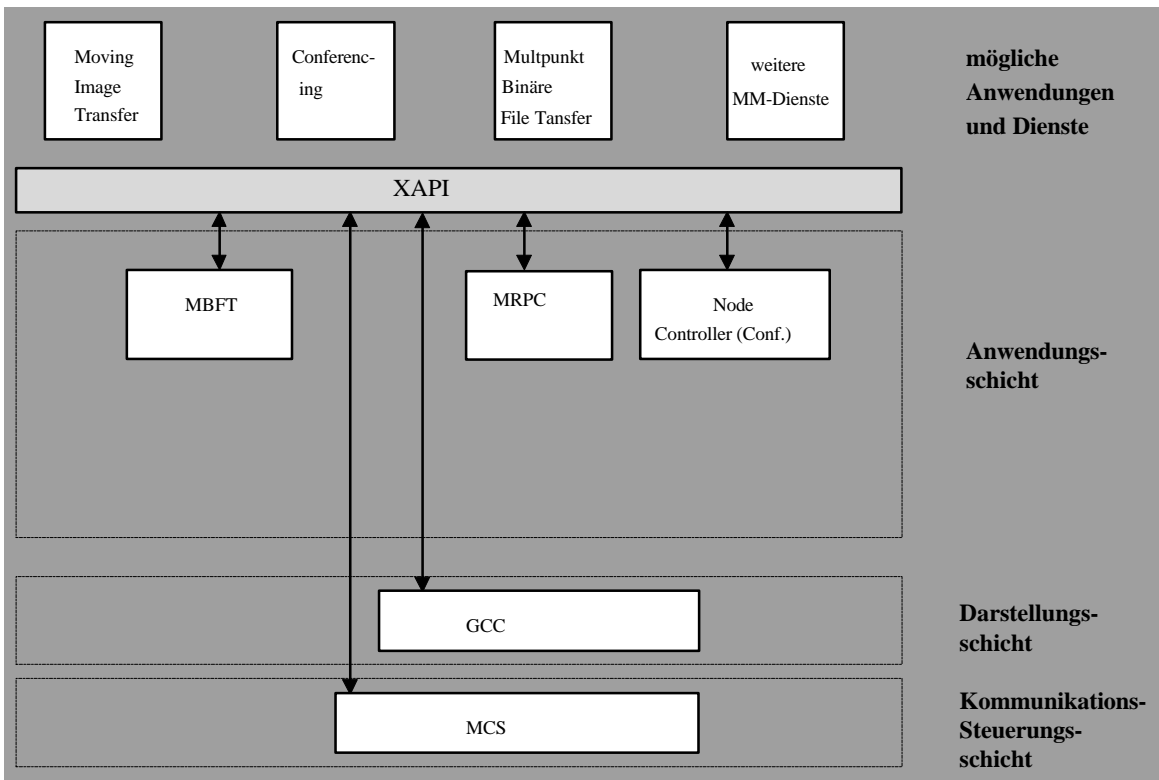


Bild 10: Zugriff der Dienste auf den Protokoll Stack für T.120

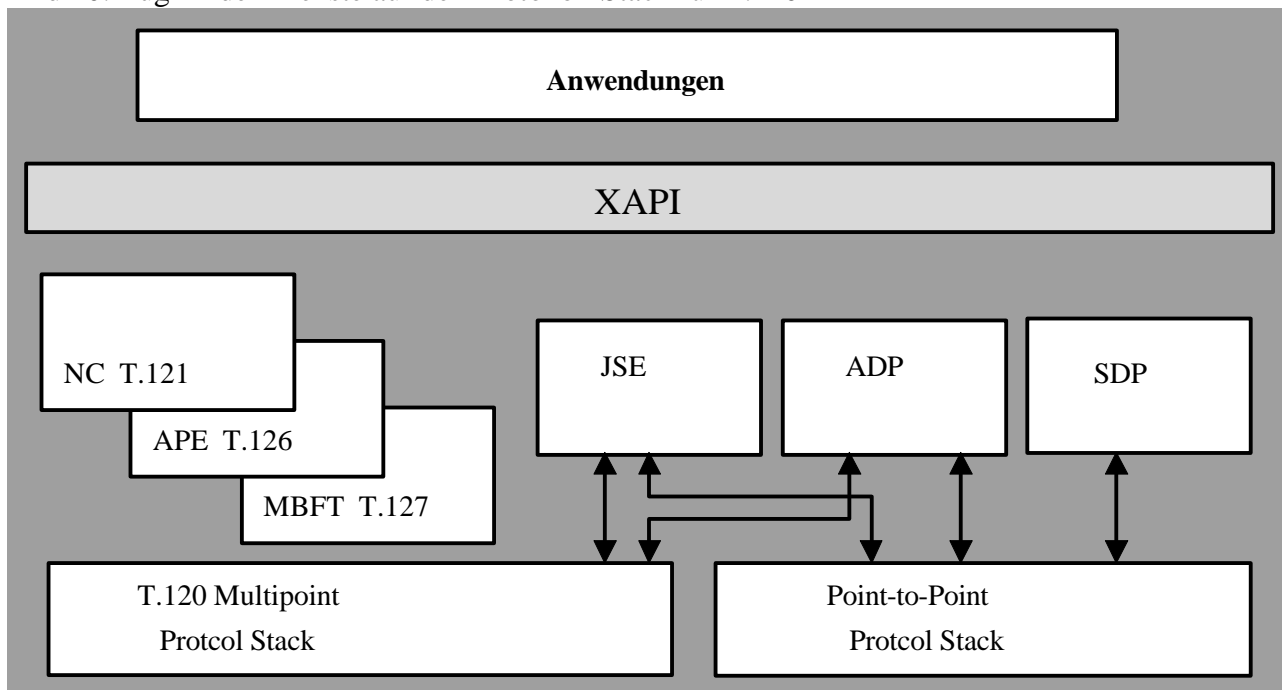


Bild 11: T.XAPI Konzept mit T.120- und Dokumentenanwendungen

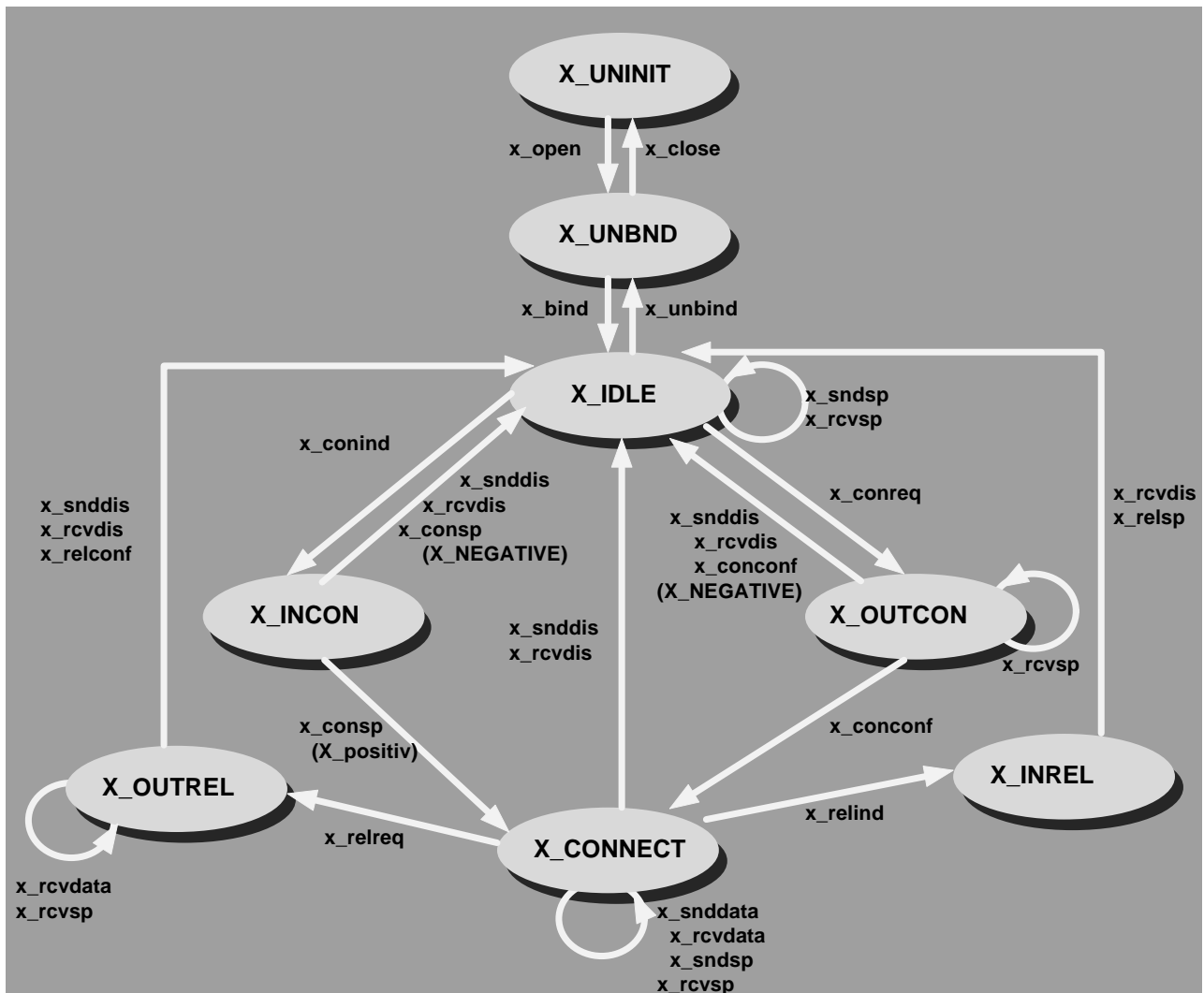


Bild 12: Status Modell für T.120

10.3 T.XAPI

Die T.XAPI wird neben dem allgemeinen architektonischen Teil ganz wesentlich auf die Belange der Mehrpunktkommunikation und auf die Anwendungen aus dem Cooperative Document Handling abgestimmt sein. So soll die T.XAPI einen einheitlichen Zugriff auf den Multipoint Stack des Standards T.120 und auf den bisher in Dokumentenanwendungen benutzten OSI Stack bereitstellen (Bild 11). So befinden sich unterhalb der XAPI die Dienste der T.120 und des Dokumenten Handling. Diese Dienste sind Joint Synchronous Editing (JSE), Asynchronous Documenten Production (ADP) und Sequential Document Production (SDP). Die Arbeiten sind noch nicht abgeschlossen und verabschiedet. Das Status Model für T.120 ist im Bild 12 anschaulich dargestellt.

10.4 DAVIC

Das Digital Audio-Visual Council (DAVIC) ist das größte Firmenkonsortium im Bereich Multimedia, das aus den bedeutendsten 150 Firmen aus aller Welt besteht. Es sollte bis Anfang 96 ein offenes System spezifizieren, so daß Inhaltersteller, Serveranbieter, Endsystemhersteller und Netzbetreiber weltweit zusammenarbeiten können. Dabei sollten Verteildienste und Aufrufdienste wie Video-on-Demand be-

rücksichtigt werden. Jetzt in der zweiten Phase sollen symmetrische Dienste wie z. B. Videokonferenz hinzukommen.

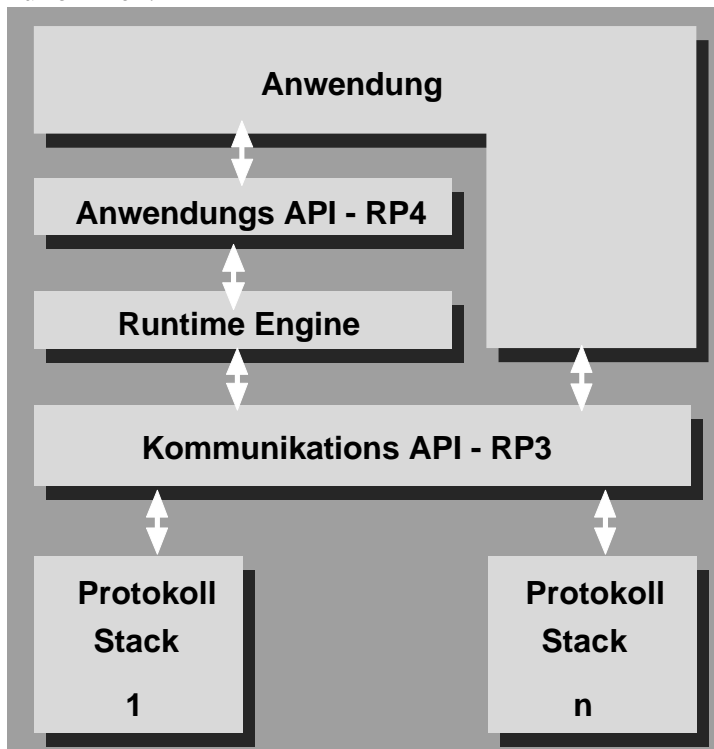


Bild 13: Modell der Set-Top-Unit bei DAVIC

In dem Themenbereich Set-Top-Unit bzw. Set-Top-Boxen wurde die XAPI als einziger Vorschlag einer universellen Kommunikationsschnittstelle diskutiert. Das Endgerät Set-Top-Unit (STU) ist für den Endanwender gedacht, um Dienste abzurufen. Es wurde beschlossen, ein Mapping mit den Protokollen für die vorhergehende Version DAVIC 1.0 z.B.: mit (TCP, RCP, Q2931) und den voraussichtlichen neuen Protokollen für die neuen Dienste in DAVIC 1.X mit e.G. T.120 durchzuführen.

Die XAPI würde dabei die Communication Appliation Programming Interface (API) mit dem Referenz Punkt 3 im Referenz Modell der STU übernehmen (Bild 13). Funktional liegt sie zwischen der Anwendung und dem Netz. Sie bietet die Kommunikation des Inhaltes und eine Kontrolle der Daten der darüberliegenden Schicht. Sie bietet weiterhin die Kommunikation zur Gegenstelle mit level 1 gateway und / oder dem Service Provider. Hier bietet sie eine Schnittstelle an, die generisch und unabhängig von Netzwerken ist.

10.5 Forschungsprojekt Audio-on-Demand

An dem Forschungszentrum der Deutschen Telekom AG entsteht ein Projekt, das sich mit dem Dienst der Audio-on-Demand beschäftigt. Der Dienst ermöglicht zwanzig Benutzergruppen, unabhängig den Dienst zu nutzen. Dieser Dienst wird über T-Online zur Verfügung gestellt und wurde auf der Messe C-Bit '96 erstmals vorgestellt. Das System basiert auf einer Workstation der Firma SUN mit dem Betriebs-

system Solaris. Die XAPI-Schnittstelle ist hier die Kommunikationsschnittstelle. Darunter befinden sich die Schnittstelle zur CAPI und des ISDN's.

10.6 XAPI für Windows 3.1

Die Deutsche Telekom AG realisierte Anfang 1996 eine Version der XAPI für Windows 3.1, die den Service Provider eines H.320 Codes auf Basis von ISDN beinhaltet. Im Entwicklungszentrum Berlin der Deutschen Telekom AG entstanden eine Testapplikation, um die XAPI unter Windows in ihren Funktionalitäten und der Robustheit zu überprüfen. Auch mit einer Realisierung einer Anwendung für Videokonferenzen mit dem Service Provider H.320 unter ISDN wurde begonnen.

Der H.320 Service Provider soll in dem Projekt der ITU-T T.120 eingesetzt werden. Weiterhin ist beabsichtigt, daß im Projekt der Europäischen Gemeinschaft EURESCOM P505 die Demonstrationsanwendung zur Videokonferenz mit dem Service Provider H.320 einzusetzen ist.

11. Fazit

Diese Entwicklung bringt uns ein Stück näher an die universelle Kommunikationsplattform, die die Erstellung neuer Dienste vereinfacht.

Die XAPI ist eine universelle Kommunikationsplattform für Datenübertragung jeglicher Art. Dabei werden auch schwierige Multimediadaten für Dienste und Anwendungen übertragen. Das besondere bei dieser Plattform ist:

- Ein homogener Zugriff zu unterschiedlichen Protokollen, sei es OSI oder Nicht-OSI-konforme Protokolle.
- Ein homogener Zugriff zu unterschiedlichen Netzwerken.
- Ein homogener Adressierungsmechanismus für unterschiedliche Netzwerke.
- Ein Integrieren von unterschiedliche Netzwerkcharakteren .
- Eine homogene Schnittstelle zur Applikation und den Diensten.
- Ein Integrieren von neuen, unterschiedlichen Protokollen und Netzwerken.
- Ein Skalieren nach den Anforderungen eines Dienstes.

12. Abkürzungen

ACSE	Association Control Service Element
BFT	Binary File Transfer
B-ISDN	Breitband-ISDN
DAVIC	Digital Audio-Visual Council
ETSI	Europäisches Institut für Standardisierung
FAX	Faksimile
ISDN	Integrated Service Digital Network
ISO	International Organization for Standardisation
ITU	International Telecommunication Union
LAN	Local Area Network

MM	Multimedia
OSI	Open System Interconnection
PDU	Protocol Data Unit
PSTN	Pubic Service Telecommunicationnetwork
ROSE	Remote Operation Service Element
RTSE	Reliable Transfer Service Element
SDU	Service Data Unit
STU	Set-Top-Unit
TCP/IP	Transmission Control Protocol / Internet Protocol
XAPI	EXtensive Application Programming Interface
XTI	X/Open Transport Interface