Universität Kaiserslautern

- AG Rechnernetze -

Entwurf eines Telefon-vermittlungssystems in Estelle

Jörg Illerich

Betreuer: Jan Bredereke

Oktober 95

Inhaltsverzeichnis

1 Einführung

Bei dem Entwurf von Kommunikationssystemen, z.B. Telefonvermittlungssystemen, ist es vorteilhaft, wenn man zuerst einen Kerndienst definiert und diesen dann um zusätzliche Leistungsmerkmale (Features) erweitert, da der Kerndienst dann bereits verifiziert werden kann, bevor Erweiterungen hinzugefügt werden. Die Trennung führt darüberhinaus zu kleinen überschaubaren Einheiten, so daß eine Erweiterung deutlich vereinfacht wird.
Zum Beispiel bei amerikanischen Telefonvermittlungssystemen kann der Kunde heute schon aus einer größen Anzahl von Features auswählen, die er einzeln zusätzlich kaufen kann. Natürlich können sich die neuen Funktionen gegenseitig stören, und da es sehr viele sind, ist es unmöglich, alle Kombinationen erschöpfend zu testen. Diese 'Feature Interactions' sind ein ernstes Problem. Mit dem Einsatz einer 'Formalen Beschreibungstechnik' (FDT), zum Beispiel Estelle, wird es möglich, das Problem auf einer abstrakteren Ebene und nach formalen Kriterien zu behandeln. Aktuelle Untersuchungen in der AG Rechnernetze zielen nun darauf, wie man ein Kommunikationssystem aus einem Kerndienst und vielen Erweiterungen derart in Estelle spezifizieren kann, daß (bestimmte Arten von) Interaktionen zwischen den Erweiterungen möglichst automatisch erkannt werden können oder eventuell auch gar nicht erst auftreten.

In dieser Projektarbeit sollte ein stark vereinfachtes Telefonvermittlungssystem in Estelle spezifiziert werden. Die Spezifikation sollte so aufgebaut sein, daß beliebige weitere Features angefügt werden können, um Methoden zur Erkennung von Feature-Interaktionen erproben zu können. So war es insbesondere wichtig, in geeigneter Weise Punkte vorzusehen, an denen später die weiteren Features angefügt werden können. Das Vermittlungssystem sollte auf den Standards für das Diensteintegrierende Digitale Netzwerk (ISDN) und für die Intelligenten Netzwerke (IN) basieren. Letztere beschäftigen sich mit den Möglichkeiten, wie Vermittlungsssyteme für neue Features geöffnet werden können. Die gesamte Spezifikation sollte durch automatische Werkzeuge analysierbar sein können. Mit Hilfe von Werkzeugen soll dann später untersucht werden, ob es zu unerwünschter Beeinflussung von Features untereinander kommen kann. Letzteres ist aber nicht mehr Bestandteil dieser Projektarbeit.

In Kapitel 2 werden die Begriffe IN, Features, ISDN erläutert. Nach dieser Einführung in Telefonvermittlungssysteme wird in Kapitel 3 die in dieser Projektarbeit in Estelle erstellte Spezifikation eines Telefonvermittlungssystemes selbst beschrieben, die Entwurfsentscheidungen und der Aufbau der Spezifikation. Schließlich wird in Kapitel 3.6 erläutert, wie diese Spezifikation um Leistungsmerkmale erweitert werden kann und mit welchen Problemen dabei zu rechnen ist. Die Erweiterungen um Leistungsmerkmale werden in zukünftigen Arbeiten vorgenommen.

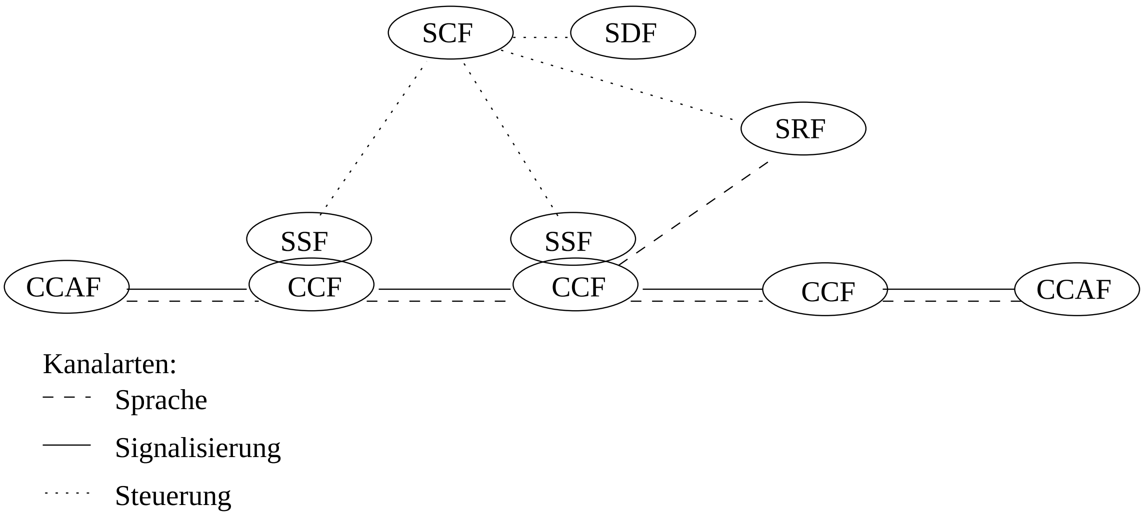
2 Grundlagen und Begriffserklärungen

2.1 Intelligente Netzwerke

Bei dem von der ITU-T standardisierten Modell der *intelligenten Netzwerke* (IN) [ITU93b, DuVi92] handelt es sich um eine Abstraktion existierender Telefonvermittlungssysteme.
Die Ziele von intelligenten Netzwerken sind:

- In Telekommunikationsnetzwerken die Service-Implementation zu erleichtern und zu beschleunigen.
- Einen herstellerunabhängigen Standard zu schaffen, wodurch u.a. eine Öffnung für Drittanbieter ermöglicht wird.
- Das Dienste-Angebot um Informationsdienste, Breitband- und Multimedia-Fähigkeiten zu erweitern.
- Sicherzustellen, daß die Dienste bei einem Netzwerkbetreiber korrekt und konsistent mit jeder Art von Ausstattung beim Kunden zusammenarbeiten.
- Weiterentwicklung existierender Netzwerke; die neue Technologie muß mit den existierenden Netzwerken zusammen laufen können, da diese nicht über Nacht ausgetauscht werden können.
- Beschreibung von Gemeinsamkeiten zwischen unterschiedlichen Systemen.

Für den Entwurf der intelligenten Netzwerke wurde das IN Begriffsmodell definiert. Dieses Modell besteht aus vier Ebenen, die die unterschiedlichen Abstraktionsebenen in IN darstellen.
Die *Dienstebene* (Service Plane) repräsentiert eine dienstorientierte Sicht von intelligenten Netzwerken, welche keine Informationen über die Implementierung der Netzwerkdienste beinhaltet.
In der *globalen funktionalen Ebene* (Global Functional Plane) sind die unterschiedlichen Funktionen eines intelligenten Netzwerkes sichtbar. In dieser Ebene befinden sich das Anrufbearbeite-Modell und die dienstunabhängigen Funktionen (service independent building blocks, SIBs), die dem Erzeugen neuer Dienste dienen, die aus einer Kombination der SIBs zusammengesetzt werden.
Die *verteilte funktionale Ebene* (Distributed Functional Plane) bietet eine Sicht auf die Verteilung im intelligenten Netzwerk. Hier werden die funktionalen Objekte, deren Aktionen und deren Beziehungen zu bzw. Informationsaustausch mit anderen Objekten beschrieben.
In der *physikalische Ebene* (Physical Plane) werden die physikalischen Aspekte des intelligenten Netzwerks betrachtet. Dazu gehören die Verteilung auf reale



CCAF	Call Control Agent Function	SDF	Service Data Function
CCF	Call Control Function	SRF	Specialized Resources Function
SCF	Service Control Function	SSF	Service Switchting Function

Abbildung 1: IN Modell der verteilten funktionalen Ebene

Baugruppen, Schaltschränke und Rechner, sowie die Kommunikationsprotokolle.

Die verteilte funktionale Ebene wird in [ITU93b] noch genauer beschrieben. Folgende funktionale Objekte gibt es auf dieser Ebene:
Die *Call Control Agent Function* (CCAF) stellt den Zugang zum Netzwerk für den Benutzer bereit und dient als Interface zwischen dem Benutzer und den Netzwerk-Anrufkontrollfunktionen. Die *Call Control Function* (CCF) ist für die Anrufübertragung und deren Kontrolle zuständig. Die *Service Switching Function* (SSF) stellt eine Menge von Funktionen bereit, die für die Interaktion der CCF und der Service Control Function (SCF) benötigt werden. Die *Service Control Function* (SCF) beinhaltet die IN Dienstlogik und steuert die Bearbeitung von Dienstaktivitäten (z.B. Analyse, Übersetzung, Routing). Je nach Bedarf wendet sich die SCF dazu an die SDF und die SRF. Die *Service Data Function* (SDF) steuert den Zugriff auf dienstbezogene und Netzwerk-Daten durch die SCF und verwaltet diese Daten auch. Die *Specialized Resources Function* (SRF) stellt spezielle Ressourcen (z.B. Spracherkennungsmöglichkeiten, Konferenzbrücken usw.) zur Verfügung, die zur Ausführung der von IN bereitgestellten Dienste benötigt werden. Abbildung 1 zeigt die Verbindungen zwischen den einzelnen Funktionen, wobei auch alle möglichen Verbindungsarten der CCF bzw. SSF mit den Dienstfunktionen (SSF,SCF,SRF) angedeutet werden.

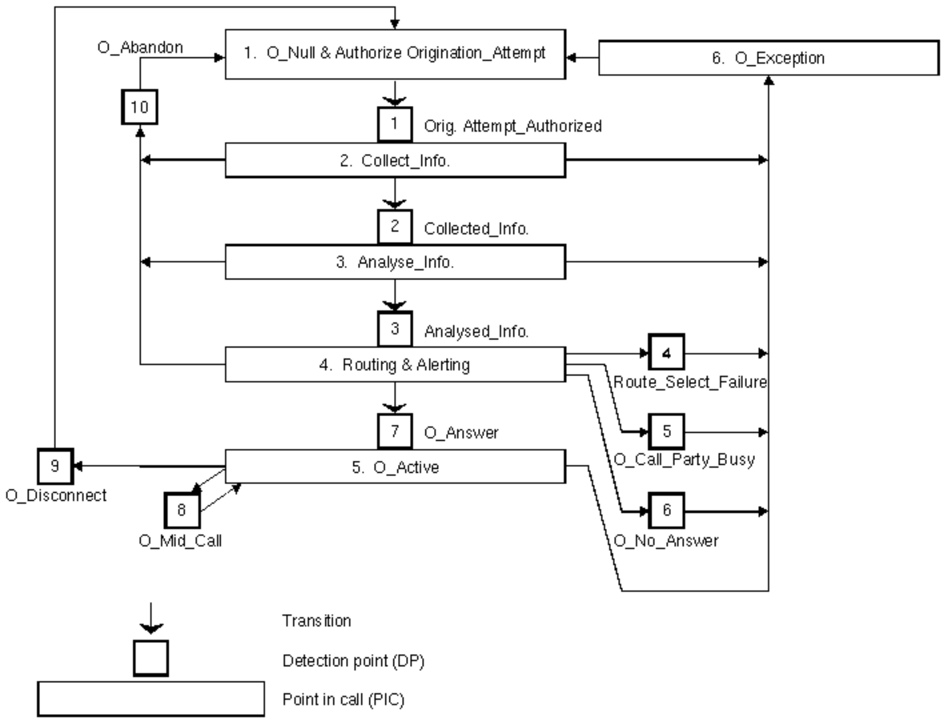


Abbildung 2: Originating call state model

In dieser Projektarbeit wurde das Telefonvermittlungssystem auf der verteilten funktionalen Ebene modelliert.
Beim Entwurf der IN-Architektur erfolgte eine Trennung in das *Basic Call Model* (BCM), welches durch ein Paar von kommunizierenden endlichen Automaten modelliert wird, und Erweiterungen. Das Basic Call Model setzt sich aus zwei unterschiedlichen Protokollmaschinen für die Anrufbearbeitelogik zusammen, nämlich eine für die Anruferseite (*originating call state model*, Abb. 2) und eine für die Angerufenenseite (*terminating call state model*, Abb. 3). Das BCM stellt die kleinste Gemeinsamkeit bereits existierender Systeme dar.

Jede Seite des Basic Call Model enthält eine Anzahl von Zuständen (Points in Call, PICs und detection points, DPs) und Transitionen, welche die unterschiedlichen Zustände illustrieren sollen, die bei einem Anruf durchlaufen werden. Die PICs stellen dabei die wesentlichen Verarbeitungsschritte während eines Anrufes dar. Sie entsprechen in etwa einem Zustand eines endlichen Automaten. Auf einer niedrigeren Abstraktionsebene setzen sich die PICs aus mehreren Einzelzuständen zusammen. Auch DPs sind in etwa mit einem Zustand eines endlichen Automaten vergleichbar. Zu den Detection Points wird bei bestimmten Ereignissen gesprungen, wie das Eintreffen einer bestimmten Nachricht. Die DPs können aktiviert (armed) werden, dann wird der IN Servicelogik (in der SCF) das Er-

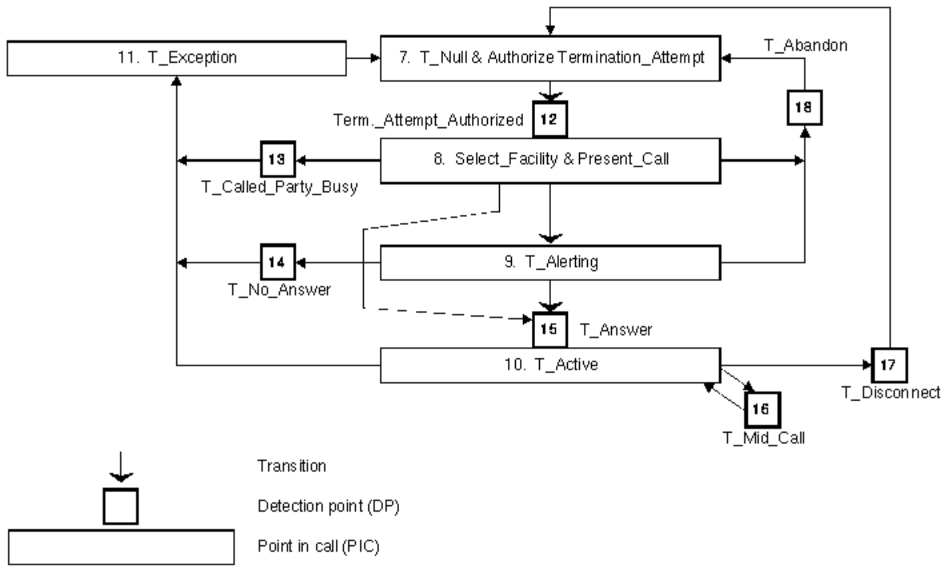


Abbildung 3: Terminating call state model

reichen eines DPs von der SSF mitgeteilt und sie (die SCF) kann somit darauf reagieren. Dieses Aktivieren eines DPs kann an eine Bedingung geknüpft werden, so daß nur bei Erfüllung dieser Bedingung eine Mitteilung an die IN Servicelogik erfolgt. Erweiterungen der Servicelogik über das BCM hinaus werden in der SCF vorgenommen, zusätzlich sind an den Detection Points entsprechende Auslösebedingungen für die Featurebearbeitung vorzunehmen.

2.2 Features

Das Grundkonzept von IN sieht vor, daß aufbauend auf dem Basisdienst jedem Teilnehmer eine individuelle Auswahl aus einer großen Menge verschiedener Kommunikationsleistungsmerkmale (*Features*) angeboten werden können sollen.
Eine genauere Beschreibung solcher Features findet sich im ITU-T Standard Q.1211 [ITU93b], hier sollen nur ein paar Beispiele für solche Features aufgelistet werden:

- Anrufweiterleitung (call forwarding):
  Ein Teilnehmer kann an ihn gerichtete Anrufe an eine andere Telefonnummer weiterleiten lassen, weil er sich z.B. gerade am zu dieser Telefonnummer zugehörigen Ort befindet. Ob die Anrufweiterleitung aktiv sein soll oder nicht, entscheidet der Teilnehmer selbst.

- Anklopfen (call waiting):
  Versucht jemand einen Teilnehmer, der bereits ein Telefonat führt, anzurufen, erhältet der Angerufene eine Mitteilung in sein Telefonat hinein, daß

jemand anderes versucht, ihn zu erreichen.

- Rückfragen (consulation calling):
  Ein laufendes Gespräch kann in Wartestellung versetzt werden, um zwischendurch zum Zwecke eines Nachfragens ein anderes Telefonat zu führen und anschließend das erste Gespräch wieder aufzunehmen.

- Schwarze Liste für unerwünschte Anrufer (Terminating call screening):
  Ein Teilnehmer kann in einer Liste angeben, von welchen Telefonnummern aus er nicht angerufen werden darf. Versucht dann jemand mit einer Telefonnummer dieser Liste den Teilnehmer anzurufen, wird er abgewiesen, ohne daß ein Verbindungsaufbau zustande kam.

- Anzeige der Telefonnummer des Anrufers:
  Bereits beim Klingeln des Telefons wird im einem Display die Telefonnummer des Anrufers angezeigt. Zusammen mit einem Computer ließe sich so der Datensatz (Name und Adresse des Anrufers, etc.) auf dem Bildschirm darstellen, bevor der Telefonhörer abgenommen wird. (Dies ist kein ITU-T Feature, da es spezielle Endgeräte mit einem Anzeigenfeld voraussetzt. Es existiert z.B. bei ISDN und bei Funktelefondiensten.)

Die Erweiterungmöglichkeiten der in dieser Projektarbeit erstellten Spezifikation um Features wird in Kapitel 3.6 beschrieben.

2.3 ISDN

Nach der Darstellung des IN Modells soll nachfolgend ein konkretes Beispiel für ein Telefonvermittlungssystem aufgeführt werden. Hier wurde das Beispiel ISDN gewählt, da dort bereits heute einige der im vorangegangenen Kapitel erwähnten Leistungsmerkmale implementiert worden sind.
ISDN steht für Integrated Services Digital Network (dienstintegrierendes digitales Netzwerk). Laut Tanenbaum [Tan92] ist das hauptsächliche Ziel von ISDN die Zusammenlegung von Sprachübertragung und anderen Diensten, die mit der Sprachübertragung nichts zu tun haben. Dazu bietet ein ISDN Basisanschluß zwei B-Kanäle, bei denen es sich um digitale PCM-Kanal für Sprache oder Daten mit einer Übertragungsrate von je 64 Kbps handelt und einen separaten D-Kanal, über den die Signalisierung mit 16 Kbps erfolgt.
Der Inhalt der B-Kanäle wird von ISDN nicht genauer spezifiziert. ISDN legt für die B-Kanäle lediglich die Bitübertragungsschicht fest. Wenn ein ISDN-Anwender mit einem anderen ISDN-Anwender Verbindung aufnimmt, können die beiden ihren Kanal in beliebige Rahmen einteilen.
Der D-Kanal wird von den Anwendern zur Kommunikation mit ISDN selbst benutzt. Verbindungen werden durch Abschicken einer Nachricht auf dem D-Kanal angefordert. Das genaue Format dieser Nachrichten wird im ITU-T Standard Q.931 [ITU93a] beschrieben. Eine typische Nachricht auf dem D-Kanal zum

Verbindungsaufbau würde den zu benutzenden B-Kanal, die anzurufende ISDN-Nummer und weitere Angaben (z.B. R-Gespräch) übermitteln.

Als typische Anwendung von zwei Kanälen nennt Tanenbaum die Möglichkeit für zwei Gesprächspartner sich per Telefon zu unterhalten und auf dem zweiten Kanal währenddessen Dokumente auszutauschen.
Die Grundidee hinter ISDN ist die der Bit-Pipeline, eine gedachte Pipeline zwischen dem Kunden und dem Netzwerkbetreiber bzw. zwischen zwei Kunden, durch die die Bits fließen. Es ist unwichtig, ob die Bits von einem digitalen Telefon, einem digitalen Terminal, einem digitalen Fernkopierer oder einem anderen Gerät kommen. Es ist lediglich von Bedeutung, daß durch die Pipeline Bits in beiden Richtungen fließen können.

Vergleicht man ISDN mit IN, so fallen die unterschiedlichen Zielsetzungen auf: ISDN hat als hauptsächliches Ziel die Übertragung von Sprache und anderen Diensten, die mit der Sprachübertragung nichts zu tun haben, dazu gibt es hier die beiden B-Kanäle. Bei IN will man dagegen den Grunddienst von existierenden Telefonvermittlungssystemen nachbilden (somit auch von ISDN), es geht hierbei hauptsächlich um den Verbindungsauf- und -abbau. Zusätzlich wurde beim Entwurf von IN besonders darauf geachtet, daß Erweiterungen über den Grunddienst hinaus auf einfache Weise vorgenommen werden können. Dazu stehen bei IN die SCF und die Detection Points zur Verfügung. Bei ISDN sind solche Erweiterungsmöglichkeiten nicht ausdrücklich vorgesehen.

3 Beschreibung der erstellten Spezifikation

Nach der Einführung in Telefonvermittlungssysteme wird in diesem Kapitel die in dieser Projektarbeit in Estelle erstellte Spezifikation eines Telefonvermittlungssystemes erläutert.

3.1 Entwurfsentscheidungen

Da die in dieser Projektarbeit erstellte Spezifikation, wie bereits in der Einleitung erwähnt, später um Features erweitert werden soll, um dann mit Hilfe eines automatischen Werkzeuges mögliche Feature Interaktionen aufzudecken, mußten bereits beim Entwurf der Spezifikation Einschränkungen, vorgegeben vom Analysewerkzeug, berücksichtigt werden. Insbesondere war dabei eine *statische* Verbindungsstruktur der Module wichtig, da das zur Zeit vorliegende Analysewerkzeug keine dynamische unterstützt. Mit statisch ist in diesem Zusammenhang gemeint, daß die Anzahl der Modulinstanzen und deren Verbindungen über Kanäle nach deren Initialisierung fest ist. Somit dürfen keine Module dynamisch erzeugt und wieder beendet werden. Um es dem Analysewerkzeug möglichst leicht zu machen, wurde darauf geachtet, die Architektur möglichst einfach zu halten. Daher wurde die Zahl der Endgeräte auf zwei reduziert.

Eine weitere Entwurfsentscheidung (bzw. Teil der Aufgabe) war, ISDN/DSS1 als unterliegendes Protokoll zu verwenden.

Gegenüber Abbildung 1 wurden die Trennung zwischen SCF und SSF aufgehoben und beide zusammen mit CCF zu einem einzigen Modul zusammengefaßt. Da sie das Basic Call State Model, bestehend aus Originating and Terminating Call State Model, beinhalten, wurde dieses Modul mit OBCSM & TBCSM (kurz OTBCSM) bezeichnet. In diesem Modul wurden die Modelle (Originating und Terminating Call State Model) zusammengefaßt. Da auf eine dynamische Erzeugung beider Modelle wegen des Analysewerkzeuges verzichtet werden mußte, müssen beide Modelle bereits in der Intialisierungsphase der Spezifikation erzeugt werden. Um die Verbindungsstruktur einfach zu halten, wurden die beiden Modelle in einem Modul zusammengefaßt, da sonst je nach Rolle des Endgerätes (Anrufer, Angerufener) ein entsprechendes Weiterleiten der Nachrichten an das richtige Modell über getrennte Kanäle notwendig wäre.
Jede andere Funktion aus Abbildung 1 wurde in einem eigenen Modul realisiert. Die Verbindungen zwischen den einzelnen Funktionen aus Abb. 1 wurden durch entsprechende Kanäle modelliert. Dies führt zu einer vereinfachten IN-Architektur, dargestellt in Abbildung 4.
Darüberhinaus wurden folgende weitere Vereinfachungen vorgenommen:
Auf dem Sprachkanal zwischen CCAF und OTBCSM, SRF und OTBCSM, als auch zwischen den beiden OTBCSMs kann momentan nur eine `'blabla'`-

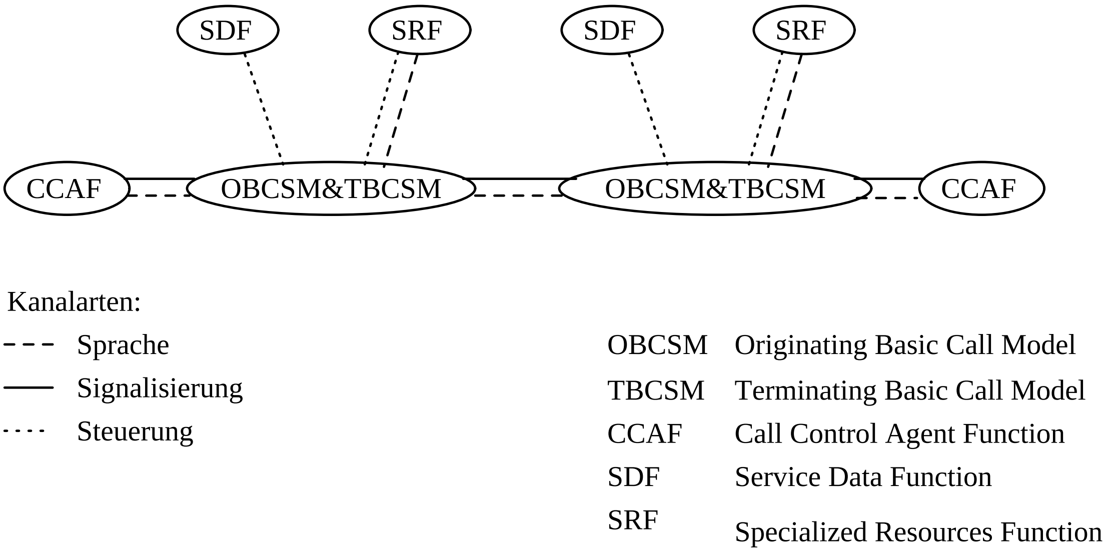


Abbildung 4: vereinfachte Systemarchitektur

Nachricht verschickt werden, da es in dieser Projektarbeit hauptsächlich um die Signale zwischen den beiden Protokollmaschinen zum Verbindungsauf- und abbau geht. Dieser Kanal kann aber natürlich später noch erweitert werden (z.B. wenn die Eingabe der Telefonnummer durch Spracheingabe ermöglicht werden soll). Gegenüber dem Standard Q.1214 [ITU93b] wurden eine vereinfachte Version der Detection Points (DP) gewählt, nämlich normale Estelle Zustände, die nur eine Durchgangsstation zum nächsten Point In Control (PIC) darstellen. Im Gegensatz zum IN-Modell müssen Erweiterungen in der erstellten Spezifikation direkt an den DPs vorgenommen werden, statt dort nur entsprechende 'Auslöser' zu plazieren, da die im IN-Modell für Erweiterungen vorgesehene Funktion SCF in der erstellten Spezifikation vom Modul OTBCSM mit übernommen werden muß. Daher ist es sinnvoll, keine Trennung zwischen dem Auslösen und der Erweiterungsbearbeitung selbst mehr vorzunehmen (wie es im IN-Modell gemacht wird).

Zur Zeit sind nur einstellige Telefonnummern vergeben, nämlich die '1' für Endgerät 1 und die '2' für Endgerät 2. Grundsätzlich ist aber eine Erweiterung auf längere Telefonnummern bereits vorgesehen.
Die Funktion `Routing_possible(PhoneNumber:PhoneNumberType):Boolean;` prüft dann, ob es sich um eine gültige Telefonnummer handelt.
Weitere Entwurfsentscheidungen wurden zum Modul OTBCSM bezüglich seiner inneren Struktur getroffen. Diese werden im Kapitel 3.4.2 erläutert.

3.2 Aufbau der erstellten Spezifikation

Die erstellte Spezifikation besteht aus vier Modulen und fünf verschiedenen Kanälen (nur fünf, da mehrere Kanäle aus Abb. 4 mit einem Estelle-Kanal modelliert wurden). Das Modul CCAF (Call Control Agent Function) stellt die Vertretung eines einfachen Endgerätes (sprich: Telefon) im Software-System dar, das Modul OBCSM&TBCSM (Originating Basic Call State Model & Terminating Basic Call State Model, kurz OTBCSM) übernimmt die Aufgaben Verbindungsaufbau und Verbindungsabbau und zusätzlich zum IN-Standard auch Gesprächsübermittlung. Die beiden Module SDF (Service Data Function) und SRF (Specialized Resources Function) sind für spätere Erweiterungen vorgesehen, wobei im Modul SDF z.B. das Übersetzen von einer Telefonnummer in eine andere (z.B. bei 0130-Nummern) vorgenommen wird und im Modul SRF z.B. Spracherkennung und Ansagen durchgeführt werden. Eine genauere Beschreibung der Module befindet sich im Kapitel 3.4.
Die Endgeräte sind jeweils über einen Sprach- und einen Signalisierungskanal mit der OTBCSM verbunden. Die OTBCSMs haben jeweils Verbindungen zu einer eigenen Instanz der Module SDF und SRF, wobei die Verbindung zur SDF aus einem Steuerungskanal und die zur SRF aus einem Steuerungs- und Sprachkanal besteht. Untereinander sind die OTBCSMs über einen Sprachkanal und einen Signalisierungskanal verbunden. Vergleiche dazu Abbildung 4. Diese Abbildung war ein Realisierungsvorschlag in der Aufgabenstellung der Projektarbeit. Ein genauere Beschreibung der Kanäle befindet sich im nachfolgenden Kapitel.

3.3 Die Kanäle im Detail

Nachfolgend werden die Kanäle zwischen den Modulen aus Abbildung 4 erläutert.

Die Sprachkommunikation zwischen CCAF und OTBCSM, OTBCSM und SRF, als auch den beiden OTBCSMs erfolgt über je einen unidirektionalen Sprachkanal.

Der bidirektionale Signalisierungskanal zwischen CCAF und OTBCSM hat zwei unterschiedliche Rollen:

- CCAF_role: Die Nachrichten (Abnehmen, Auflegen des Telefonhöres, Tastendrücke), die das Endgerät an die OTBCSM schickt.
- OTBCSM_role: Die Nachrichten (Signaltöne), die die OTBCSM an das Endgerät schickt.

Da bei Estelle nur Interaktionspunkte mit unterschiedlichen Rollen verbunden werden können, mußte der an sich bidirektionale Signalisierungskanal zwischen den beiden OTBCSMs in zwei unidirektionale Kanäle umgewandelt werden, wobei dann eine sich überkreuzende Verbindung wie in Abbildung 5 vorgenommen wurde.

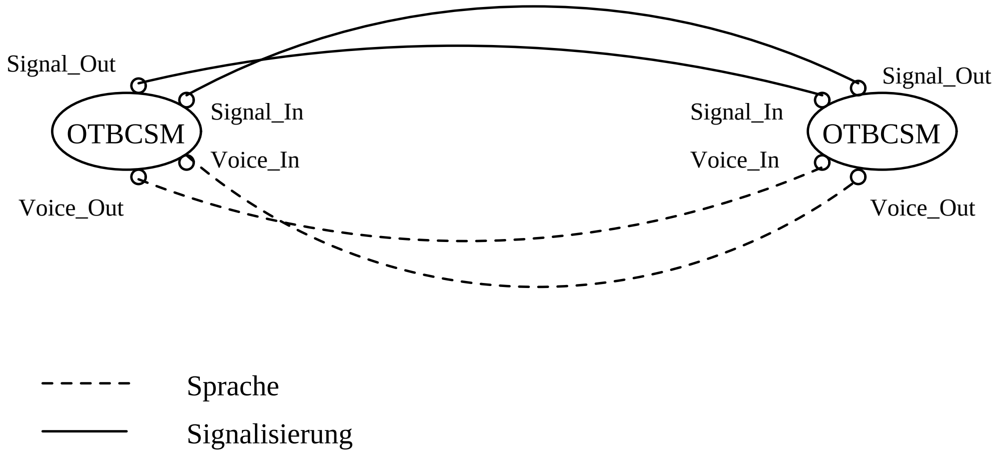


Abbildung 5: Kanäle zwischen den OTBCSMs

Auf dem Signalisierungskanal zwischen den beiden OTBCSMs werden folgende Nachrichten, die dem ITU-T Standard Q.931 [ITU93a], d.h. ISDN entnommen sind, ausgetauscht:

- `Setup`: teilt Verbindungsaufbau-Wunsch mit.
- `Alerting`: Rückmeldung, daß es klingelt.
- `Connectmsg`: Verbindung ist zustande gekommen, Gegenüber hat Hörer abgenommen.
- `Connect_acknowledge`: Bestätigung, daß die OTBCSM des Anrufers die Connectmsg erhalten hat.
- `Disconnectmsg`: Jemand hat bei bestehender Verbindung aufgelegt oder Gegenüber ist besetzt.
- `Key_pressed(Key:Keytype)`: Tastendruck auf Telefon.

Der bidrektionale Steuerungskanal zwischen OTBCSM und SDF enthält zur Zeit nur dummy-Nachrichten, die bei späteren Erweiterungen sinnvolleren Nachrichten weichen können. Das gleiche gilt für den bidrektionale Steuerungskanal zwischen OTBCSM und SRF.

3.4 Aufbau der Module im Detail

Nachfolgend wird beschrieben, wie die Module aus Abbildung 4 in der erstellten Spezifikation realisiert wurden.

Die Module SDF und SRF werden zur Zeit nur mit einer `initialize`-Transition initialisiert, ansonsten gibt es keine Transitionsübergänge. Diese Module sind für spätere Erweiterungen in die Spezifikation aufgenommen worden. Dagegen sind die Module CCAF und OTBCSM von besonderem Interesse, so daß nachfolgend auf diese Module genauer eingegangen wird.

3.4.1 Modul CCAF

Das Modul CCAF stellt die Vertretung eines einfachen Endgerätes im Softwaresystem dar. Das Erzeugen von Nachrichten des Endgerätes wird mit Hilfe einer xtpanel Oberfläche (siehe Abbildung 6) vorgenommen, welche das Absetzen von Nachrichten mittels Mausklicks auf spezielle Buttons ermöglicht. Bei xtpanel handelt es sich um ein Programm, das mittels einfacher Skriptsprache erlaubt, ein interaktives Programm zu erzeugen. Diese Oberfläche stammt bis auf einige kleinere Erweiterungen und Anpassungen ursprünglich von Jan Bredereke. Das Entgegennehmen von Nachrichten führt zu Meldungen im entsprechenden Fenster. Das Modul CCAF bekommt von der Oberfläche die Benutzeraktionen mitgeteilt und leitet diese als entsprechende Nachrichten über den Signalisierungskanal an die OTBCSM weiter. Das Entgegennehmen von Nachrichten der OTBCSM führt zu Meldungen im entsprechenden Fenster der Oberfläche.

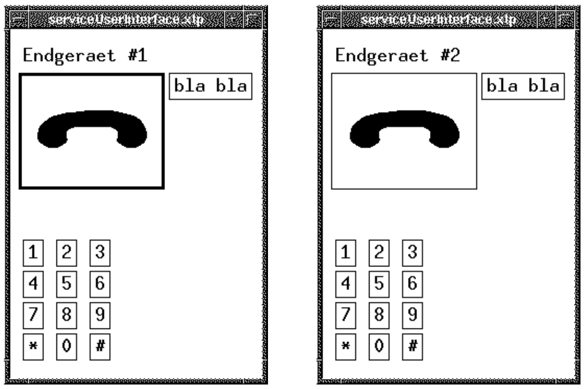


Abbildung 6: Endgerät 1 und 2

3.4.2 Modul OTBCSM

Die eigentliche Aufgabe des Moduls OTBCSM ist die Steuerung des Verbindungsauf- und abbaus. Wie im Kapitel 3.1 bereits erläutert, müssen hier zusätzlich die

Erweiterungen um Leistungsmerkmale eingefügt werden.

Die Spezifikation des Moduls OTBCSM geschah in Anlehnung an den ITU-T Standard Q.1214 [ITU93b]. Insbesondere wurden die Übergänge zwischen den PICs und DPs im *originating call state model* und dem *terminating call state model* (Abb. 2 und 3, siehe Kapitel 2.1) in Estelle-Transitionen umgesetzt.
Diese beiden Modelle wurden wie bereits in Kapitel 3.1 gesagt in einer einzigen Protokollmaschine OTBCSM zusammengefaßt, indem die Zustände `1. O_Null & Authorize Origination_Attempt` und `7. T_Null & Authorize Termination_Attempt` der beiden Einzelmodelle zu einem gemeinsamen Zustand `1. Null` überlagert wurden. Dies hat allerdings zur Folge, daß nur noch eine Modulinstanz existiert, die nicht mehr gleichzeitig beide Modelle durchlaufen kann. Dies führte zu dem folgenden Problem: Versuchen beide Teilnehmer gleichzeitig den jeweils anderen Teilnehmer anzurufen, so befinden sich beide Protokollmaschinen OTBCSM im originating call state model. Beide Protokollmaschinen müßten den Verbindungsaufbauwunsch der anderen Protokollmaschinen mit einer *busy*-Nachricht abweisen. Dies ist aber nur im terminating call state model vorgesehen, welches aber in diesem Fall nicht durchlaufen wird (siehe dazu Abb. 8). Daher mußte ein zusätzlicher Subautomat für den Fall eines gleichzeitigen Verbindungsaufbaus in das *originating call state model* hinzugefügt werden, der dann entsprechende *busy*-Nachrichten erzeugt.
Die Unterteilung in Subautomaten geschah in Anlehnung an [Bre95b], indem die Zustände des Subautomaten mit Variablen verwaltet werden.
Ein weiterer Subautomat realisiert die Gesprächsübermittlung zwischen CCAF und OTBCSM, OTBCSM und OTBCSM, OTBCSM und SRF.
Der Zustand des Subautomaten für die Gesprächsübermittlung wird über eine spezielle Hilfsvariable `connected` verwaltet, die angibt, ob von der OTBCSM eine Verbindung zur SRF oder zur anderen OTBCSM oder zur SRF der anderen OTBCSM oder von der anderen OTBCSM zu der an dieser OTBCSM angeschlossenen SRF oder zu keinen von diesen besteht.

Die Weiterleitung von Tastendrücken auf dem Telefon, sofern eine Verbindung besteht, wurde wie folgt realisiert: Zuerst wird zum `DP_MID_CALL` gesprungen und wenn dort der Tastendruck nicht verarbeitet wurde, wird er an die andere Protokollmaschine weitergeleitet. Dort kann dann in `DP_MID_CALL` der Tastendruck verarbeitet werden. Zur Zeit wird er aber dort nur entfernt.
Diese Weiterleitung stellt sicher, daß sowohl auf der Erzeugerseite (Erzeuger der Tastendrücke), als auch auf der Empfängerseite auf den Tastendruck reagiert werden kann.

Um sicherzustellen, daß die Protokollmaschinen auch bei Empfang von nicht spezifizierten Nachrichten nicht in einen Verklemmungszustand geraten, wurden Transitionen hinzugefügt, die unspezifizierten Empfang in allen vom Ersteller dieser Projektarbeit vorhergesehenen möglichen Problemsituationen verhindern, indem sie nicht erwartete Nachrichten aus der Warteschlage des Kanals entfernen.

Nachfolgend wird noch die Anordnung der Transitionen in dem Modul OTBCSM und die Bezeichnungsweise für die Zustände erläutert.
Die Angabe der Transitionen erfolgt nach Zuständen geordnet. Begonnen wurde mit dem gemeinsamen Zustand `PIC_Null`. Dann folgen die Transitionen der Zustände für das originating call state model, dann die für das terminating call state model. Dann folgt der Subautomat für die Gesprächsübermittlung und der für die Erzeugung der busy-Nachrichten bei gleichzeitigem Verbindungsaufbau.

Damit die PICs und DPs leicht in der Spezifikation wieder zu finden sind, wurde die Bezeichnung der Estelle-Zustände wie folgt gewählt:
Name des PICs oder DPs mit vorangestelltem '`PIC_`' bzw. '`DP_`'. PICs und DPs charakterisieren nur die wesentlichen Verarbeitungsschritte. Um das in der Praxis dort statt findende Verhalten wiederzugeben, muß ein PIC bzw. DP in Unterzustände unterteilt werden, da dafür normalerweise ein einzelner Zustand nicht ausreicht. Wenn ein PIC oder DP in Unterzustände aufgeteilt werden mußte, wurde an den Namen des PICs bzw. DPs ein '`_SS_`' (für substate) und eine Bezeichnung für den Unterzustand angehängt. Unterzustände wurden dabei ebenfalls als normale Estelle-Zustände realisiert, der Übergang von Haupt- zu Unterzuständen wurde dann durch entsprechende Transitionen realisiert, wie es in [Bre95b] vorgeschlagen wurde.

3.5 Zusammenspiel der Module im Detail

Nachfolgend werden anhand einiger Beispielabläufe das Zusammenspiel der Module erläutert. Dabei sind in der vorliegenden, noch nicht erweiterten, Spezifikation insbesondere der Verbindungsauf- und abbau von Interesse.

3.5.1 Verbindungsaufbau

Im nicht erweiterten System sind grundsätzlich zwei verschiedene Fälle denkbar:

1. ein erfolgreicher Verbindungsaufbau:
   Ein beispielhafter Ablauf wäre:
   Benutzer 1 ruft Benutzer 2 an, beim dem der Telefonhörer aufliegt, so daß dessen Telefon nach der Anwahl klingelt und daraufhin Benutzer 2 das Gespräch entgegennehmen kann (siehe Abb. 7).

2. ein mißglückter Verbindungsaufbau:
   Benutzer 1 erreicht Benutzer 2 nicht, da letzterer momentan den Hörer

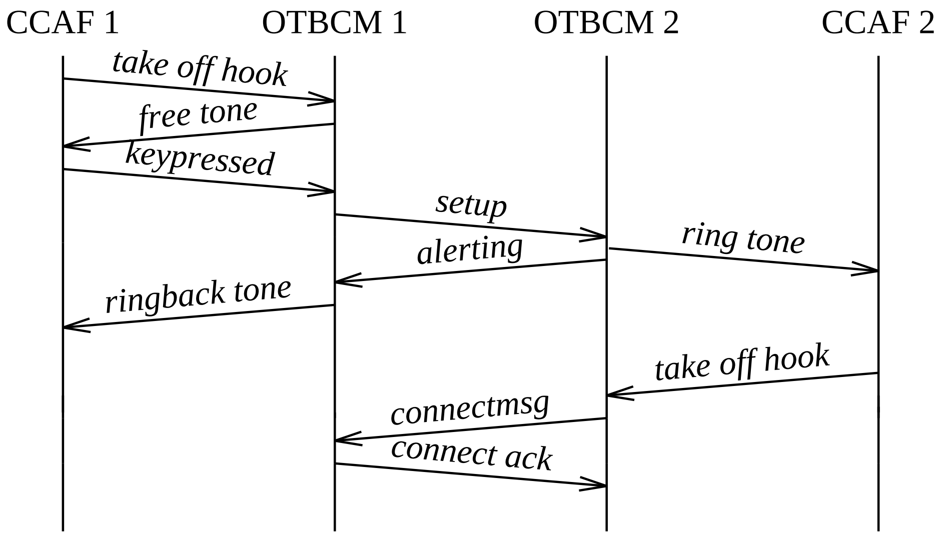


Abbildung 7: Time-Sequence Diagramm: erfolgreicher Verbindungsaufbau

abgenommen hat (weil er gerade selbst telefonieren will, oder weil er bereits ein Gespräch führt) (siehe Abb. 8).

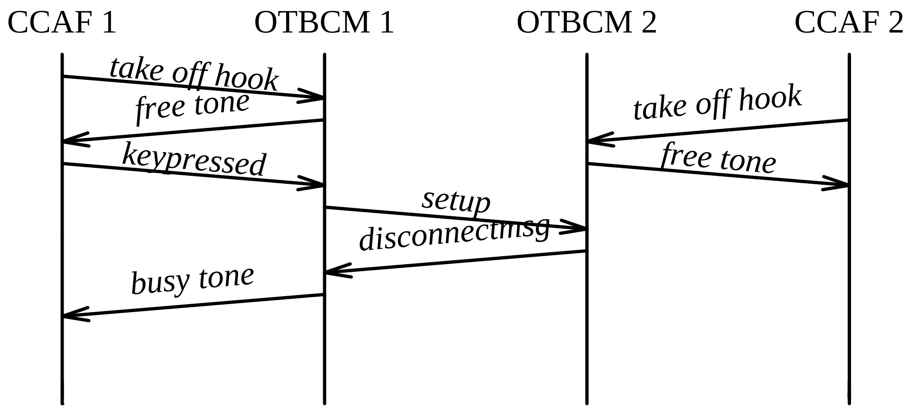


Abbildung 8: Time-Sequence Diagramm: gleichzeitiger Verbindungsaufbau

3.5.2 Verbindungsabbau

Ein möglicher Verbindungsabbau bei einer bestehenden Verbindung könnte wie folgt aussehen:

Benutzer 1 legt auf, daraufhin erhält Benutzer 2 ein akustisches Signal und er legt ebenfalls auf (siehe Abb. 9).

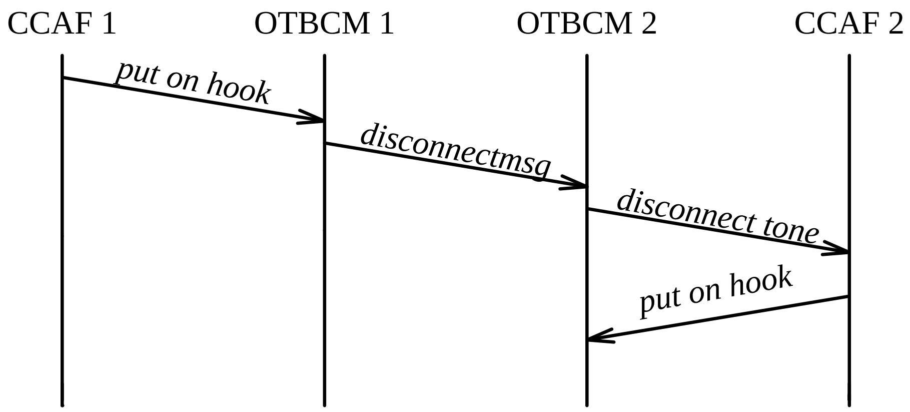


Abbildung 9: Time-Sequence Diagramm: Verbindungsabbau

3.6 Erweiterungsmöglichkeiten

Als Erweiterung der erstellten Spezifikation ist besonders die Hinzunahme von Leistungsmerkmalen von Interesse, da diese Spezifikation entworfen wurde, um nach ihrer Erweiterung um Leistungsmerkmale mit Analysewerkzeugen untersuchen zu können, ob sich Leistungsmerkmale gegenseitig beeinflussen können.

Erweiterung um Features
Neu hinzugefügtes Verhalten kann nach [Bre95a] als eine Erweiterung der Automaten des Basic Call Model realisiert werden. Die Erweiterungen werden in Estelle als Gruppen von Transitionen repräsentiert.
Für die Erweiterung an Detection Points bieten sich nach [Bre95a] zwei Möglichkeiten an:

1. Ersetzen der Transition vom DP zum PIC durch
   (a) i. eine neue Transition
       ii. durch mehrere Transitionen, die hintereinander ausgeführt den Übergang DP $\rightarrow$ PIC realisieren
       (z.B. DP $\rightarrow$ Hilfszustand1, Hilfszustand1 $\rightarrow$ PIC).
   (b) durch mehrere konkurrierende Transitionen, die alle den Übergang DP $\rightarrow$ PIC haben.
2. Anfügen dieser Erweiterung am DP selbst mit anschließender Rückkehr zum DP.

Die Möglichkeiten 1.a.i) und 1.a.ii) bereiten keine Probleme.
Beispiel zur Möglichkeit 1.a.i):

```
const  otbcmPriority_low  = 100;
       otbcmPriority_high = 110;
       feature1Priority   = 200;
```

(* unbedingter Übergang ohne Aktionen von DP → PIC in der Spezifikation ohne Erweiterungen*)

```
I)   TRANS
       PRIORITY otbcmPriority_low
       FROM DP_...
         TO PIC_...
       NAME ...
         BEGIN
         END;
```

(* wird jetzt überlagert (durch Hinzunahme einer Feature-Transition) *)

```
II)  TRANS
       PRIORITY feature1Priority
       FROM DP_...
         TO PIC_...
       NAME ...
         BEGIN
            (*   hier beliebige Aktionen je nach Feature  *)
         END;
```

Da `feature1Priority` > `otbcmPriority_low` ist, wird Transition I) durch Transition II) überlagert, da Transition II) aufgrund der höheren Priorität *immer* vor Transition I) bereit ist, so daß Transition I) nie schalten kann.

Spezieller Spezifikationsstil :
Schachelt man zusätzlich um Transition II) ein `#ifdef`, `#endif` wie folgt:

```
#ifdef feature1
 (* hierzwischen Transition II) *)
#endif
```

so ermöglicht dies ein wahlweises Hinzunehmen oder Weglassen eines Features durch entsprechende Auswahl beim Compilieren.

Der Übergang vom DP → PIC führt dazu (insbesondere auch bei Nichtdeterminismus durch Verwenden der selben Priority-Konstanten in den Erweiterungstransitionen), daß immer nur eine Erweiterung berücksichtigt wird, da nach Abarbeitung einer Erweiterungstransition sich der Automat bereits in einem anderen Zustand (`PIC_...`) befindet.

Die zweite Möglichkeit hat hier den Vorteil, daß nach Abarbeitung einer Erweiterung eine weitere Erweiterung Berücksichtigung finden kann. Nutzt man

hier Nichtdeterminismus bei den Erweiterungstransitionen aus, indem man bei diesen Transitionen die gleiche Priority-Konstante wählt, so läßt sich dadurch in der Spezifikation ausdrücken, daß die Reihenfolge der Abarbeitung offen sein soll. Da immer nur eine Erweiterungstransition zu einem Zeitpunkt feuert, ist eine atomare Ausführung für jede Erweiterung sicher gestellt, sprich es gibt keine Nebenläufigkeit zwischen den einzelnen Zuständen der Erweiterungen. Allerdings bleibt zu beachten, daß nach der Rückkehr von Transitionen einer Erweiterung zum DP diese Erweiterungstransitionen unmittelbar darauf erneut feuerbar sind (was im allgemeinen nicht erwünscht ist). Will man dies vermeiden, muß mitprotokolliert werden, welche Erweiterungszweige bereits durchlaufen wurden. Dies könnte man mit einer Hilfsvariablen pro Erweiterungszweig machen, die in der `PROVIDED`-Klausel entsprechend abgefragt wird und dann im Transition-Ausführungsteil entsprechend als abgearbeitet gesetzt wird.
Nachdem alle Erweiterungen an diesem DP durchlaufen wurden, wird dann die Transition DP $\rightarrow$ PIC feuerbar. Der Preis ist allerdings eine unschöne Erweiterung des Zustandes, denn man braucht für jeden Erweiterungszweig ein Bit, welches angibt, ob der Zweig bereits durchlaufen wurde, oder nicht.
Gegenüber der Angabe verschiedener Priority-Konstanten bei den Erweiterungstransitionen und damit der impliziten Festlegung einer Ausführungsreihenfolge bietet die Ausnutzung von Nichtdeterminismus den Vorteil, daß noch alle Verhaltensvarianten in der Spezifikation enthalten sind und lediglich die Reihenfolgen, die zu Feature-Interaktionen führen, ausgeschlossen werden müssen. Ein automatisches Werkzeug kann nun zur Erkennung von Problemsituationen die Zustände mit Nichtdeterminismus als Ausgangspunkt für die Analyse nehmen.

Ein ganz andere Möglichkeit wäre, durch nebenläufige Module den Nichtdeterminismus zu realisieren. Bei Erreichen des DPs würde man dann eine Nachricht an diese Module schicken, die dann nebenläufig arbeiten würden. Da aber auch Erweiterungen möglich sein sollen, die nach ihrer Abarbeitung zu einem anderen Rückkehrpunkt als dem normalen springen, würde die Nebenläufigkeit zum Konflikt führen, wenn zwei oder mehr Erweiterungen dies gleichzeitig erfordern würden, da dann nicht klar wäre, in welchen Zustand weiter fortgefahren werden soll. Daher erscheint diese Möglichkeit als recht unbrauchbar.

Erweiterungen an den noch deaktivierten Übergängen
Es gibt zur Zeit einige Übergänge, die mit `PROVIDED false` deaktiviert wurden, da ein Übergang zu diesen Zuständen für das System ohne Erweiterungen keinen Sinn machen.
Dazu gehören beispielsweise die Übergänge von `PIC_Collect_Info` $\rightarrow$ `PIC_O_Exception` oder `PIC_O_Active` $\rightarrow$ `PIC_O_Exception` (siehe dazu Abb. 2). Ein Übergang nach `PIC_O_Exception` wäre von diesen Zuständen aus immer möglich, wenn man den Übergang nicht an eine Bedingung knüpfen würde. Ein bedingungsloser Übergang würde aber den normalen Ablauf stören. Für spätere Erwei-

terungen wurden diese Übergänge bereits in die Spezifikation mitaufgenommen, da durch die Hinzunahme von Erweiterungen diese Übergänge notwendig werden könnten.

Routing

Unter Routing versteht man die Wegewahl für den Leitungsaufbau von der Quell- zur Zielmaschine (siehe dazu [Tan92]).
Zur Zeit findet noch kein Routing statt. Die dafür vorgesehende Prozedur `DoRouting` ist noch leer, da zwischen zwei Teilnehmern Routing noch keinen rechten Sinn macht. Eine Unterstützung dafür wurde aber für eine spätere Erweiterung auf mehr Teilnehmer schon einmal vorgesehen.

4 Zusammenfassung

Diese Projektarbeit beschäftigte sich mit Telefonvermittlungssystemen. Dazu wurden in Kapitel 2.1 die intelligenten Netzwerke vorgestellt, die eine Abstraktion existierender Telefonvermittlungssysteme darstellen. Es wurde deren Hauptziel, nämlich die leichte Erweiterbarkeit um neue Dienste, und die dazu vorgenommene Unterteilung des IN Modells in vier Ebenen (Dienstebene, globale funktionale Ebene, verteilte funktionale Ebene und physikalische Ebene) aufgeführt. Dabei wurde besonders auf die verteilte funktionale Ebene eingegangen, da das in dieser Projektarbeit in Estelle spezifizierte Telefonvermittlungssystem auf dieser Ebene modelliert wurde.
In Kapitel 2.2 wurden Beispiele für Diensterweiterungen über den Basisdienst hinaus angeführt. Danach wurde in Kapitel 2.3 auf ein konkretes Beispiel für ein Telefonvermittlungssystem, ISDN (in welchem teilweise die eben erwähnten Diensterweiterungen bereits implementiert wurden), eingegangen. Dabei wurde das hauptsächliche Ziel von ISDN, die Zusammenlegung von Sprachübertragung mit anderen Diensten, die mit Sprachübertragung nichts zu tun haben, genannt.

Nach dieser Einführung in Telefonvermittlungssysteme wurde in Kapitel 3 die in dieser Projektarbeit erstellte Spezifikation eines Telefonvermittlungssystemes vorgestellt. Es wurden die Entwurfsentscheidungen (Kapitel 3.1) insbesondere die vorgenommen Vereinfachungen und Abstraktionen erläutert. Diese Vereinfachungen waren notwendig, da die erstellte Spezifikation in zukünftigen Arbeiten um neue Leistungsmerkmale erweitert und dann mit einem Analysewerkzeug untersucht werden soll, ob es zu gegenseitigen Beeinflussungen zwischen diesen Leistungsmerkmalen (Feature Interaktionen) kommen kann.
Nach einer Beschreibung des Aufbaus der erstellten Spezifikation, einer Erläuterung der verwendeten Kanäle und Module und deren Zusammenspiel, wurde in Kapitel 3.6 erläutert, wie die Erweiterung um Leistungsmerkmalen vorgenommen werden kann und welche Probleme dabei auftreten können.

Zukünftige Arbeiten werden zeigen, inwieweit Analysewerkzeuge Feature Interaktionen erkennen können.

Literatur

[Bre95a] Bredereke, J. *Erweiterung eines EA an DPs.* Interne Notiz (Jan. 1995).

[Bre95b] Bredereke, J. *Verfeinerung von Verhalten in Estelle.* Interne Notiz (Jan. 1995).

[DuVi92] Duran, J. M. und Visser, J. *International standards for Intelligent Networks.* IEEE Commun. Mag. **30**(2),34-42 (Feb. 1992).

[ITU93a] ITU-T, Recommendation Q.931, *DSS 1 - ISDN User-Network Interface for Basic Call Control* (März 1993).

[ITU93b] ITU-T, *Q12xx-Series Intelligent Network Recommendations* (1993).

[Tan92] Tanenbaum, A. S. *Computer-Netzwerke* (1992)

5 Anhang