KERNFORSCHUNGSZENTRUM

KARLSRUHE

Oktober 1977 KFK 2526

Institut für Datenverarbeitung in der Technik

Verfahren zur Sicherung der operationalen Integrität in verteilten Datenbasen bei dezentraler Kontrollstruktur

O. Drobnik

GESELLSCHAFT
FÜR
KERNFORSCHUNG M.B.H.

KARLSRUHE

GESELLSCHAFT FÜR KERNFORSCHUNG M. B. H.

KARLSRUHE

KERNFORSCHUNGSZENTRUM KARLSRUHE

KFK 2526

Institut für Datenverarbeitung in der Technik

Verfahren zur Sicherung der operationalen Integrität in verteilten Datenbasen bei dezentraler Kontrollstruktur[+)]

O. Drobnik

Gesellschaft für Kernforschung m.b.H., Karlsruhe

[+)] von der Fakultät für Informatik der Universität Karlsruhe (Technische Hochschule) genehmigte Dissertation

Kurzfassung

Der Bericht enthält eine Fallstudie für die Entwicklung dezentralisierter Koordinierungsmechanismen in verteilten DV-Systemen. Als konkreter Anwendungsbezug dient der Problembereich der Sicherung der operationalen Integrität in verteilten Datenbasen bei parallelen Zugriffen. Die Verfahren sind insbesondere auch für die Behandlung des Ausfalls und der Wiedereingliederung von Systemkomponenten ausgelegt. Technische Lösungen zur Realisierung der Verfahren unter Verwendung neuer Systemtechnologien werden anhand der Ergebnisse von Experimenten mit realitätsnahen Simulationsmodellen diskutiert.

Decentralized control of parallel access to distributed databases

Abstract

Decentralized coordination mechanisms are developed and shown to provide operational integrity for a distributed data base under parallel access. The mechanisms include procedures to handle failure and reintegration of system components. Two different approaches to the implementation of coordination mechanisms are compared by means of simulation experiments.

Inhaltsverzeichnis Seite

Seite

1. Problemstellung

Die Hauptrichtung der Forschung und Entwicklung auf dem Gebiet der verteilten DV-Systeme (Rechnernetze, Mehrrechnersysteme usw.) zielte bisher auf die Schaffung leistungsfähiger Hardware- und Softwarekomponenten für die Interrechnerkommunikation. Der hier erreichte hohe technische Stand /H5,K2,S5,S6/ bietet mittlerweile eine ausreichende Grundlage für die Behandlung übergeordneter Problemstellungen.

Die erweiterte Verfügbarmachung vorhandener Betriebsmittel als Hauptargument für den Aufbau verteilter DV-Systeme stellt die dezentral organisierte Verwaltung von Betriebsmitteln als übergeordnete Problemstellung von fundamentaler Bedeutung heraus.

Die vorliegende Arbeit nimmt sich dieser Problemstellung an. Als konkreter Anwendungsbezug dient das ständig an Beachtung gewinnende Gebiet der Datenbanktechnologie: Der Aufbau verteilter Datenbasen verlangt vordringlich nach leistungsfähigen dezentralen Kontrollmechanismen zur Betriebsmittelvergabe und koordinierten Bearbeitung von Aufträgen. Durch den Einsatz derartiger Mechanismen wird die optimale Nutzung der Möglichkeiten verteilter DV-Systeme bei der Bearbeitung komplexer zusammenhängender Aufträge, wie sie datenbasisorientierte Transaktionen oder Prozeßlenkungsaufgaben darstellen, überhaupt erst erreichbar.

Vorgehensweisen zur zentral kontrollierten Auftragsvergabe und Betriebsmittelzuweisung, bei der genau eine Kontrollinstanz alle Aufträge und Betriebsmittel im verteilten DV-System verwaltet, sind unter Berücksichtigung verschiedenartiger Optimierungskriterien, wie z.B. die Minimierung der Antwortzeit, in /H3/ diskutiert.

Gegenüber der zentralen Kontrollstruktur bietet eine dezentralisierte Kontrolle wesentliche Vorteile, wie

- Erhöhung der Zuverlässigkeit und Verfügbarkeit des Gesamtsystems,
- Verbesserung des Antwortzeitverhaltens des Gesamtsystems durch parallele Auftragsvergabe,
- Reduktion der Systembelastung, insbesondere der Datenflüsse,

- Anpaßbarkeit der Kontrollstruktur an die Organisationsform kompliziert gegliederter Aufgabenbereiche.

Zur optimalen dezentral organisierten Auftragsvergabe werden Koordinierungsmechanismen benötigt, für die erst für begrenzte Problembereiche, wie z.B. das Multi-Kopien-Problem bei verteilten Dateien, Lösungsansätze existieren.

Eine dezentrale Kontrollstruktur - siehe Bild 1.1 - basiert auf einem System funktionell äquivalenter, gleichberechtigter Kontrollinstanzen, die Nachrichten austauschen können. Jeder Kontrollinstanz ist eine Menge von Betriebsmitteln zugeordnet, über deren Zuweisung an Aufträge sie entscheidet. Jede Kontrollinstanz kann Aufträge entgegennehmen; ein Auftrag kann sich bei einer oder bei mehreren Kontrollinstanzen um Bearbeitung bewerben. Die Betriebsmittelzuweisung nimmt eine Kontrollinstanz anhand von Zustandsinformation über die Betriebsmittel vor. Zur Entscheidung, welcher Kontrollinstanz die Überwachung der Auftragsbearbeitung zu übertragen ist, führt eine Kontrollinstanz auch Information über fremde Betriebsmittel.

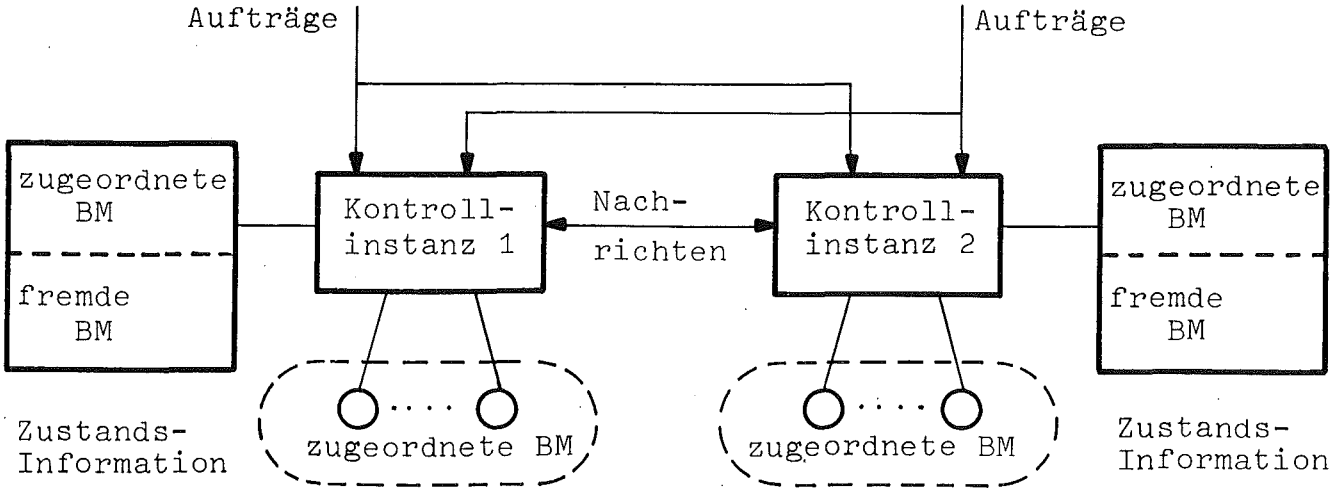


Bild 1.1: Prinzip der dezentralisierten Kontrolle der Auftragsvergabe und der Betriebsmittelzuweisung in verteilten DV-Systemen (BM = Betriebsmittel)

Die Kontrollinstanzen müssen Nachrichten austauschen, z.B.

- zur Übergabe von Aufträgen an andere Kontrollinstanzen,

- zur koordinierten Betriebsmittelzuweisung an Aufträge, die zu ihrer Ausführung gleichzeitig Betriebsmittel mehrerer Kontrollinstanzen benötigen,
- bei Ausfall von Betriebsmitteln oder Kontrollinstanzen zur Überführung des gestörten Systems in ein funktionsfähiges Restsystem.

Der Nachrichtenaustausch hat nach vorgegebenen, von allen Kontrollinstanzen zu beachtenden Regeln, sog. Kommunikationsprotokollen /G1,L2/, zu geschehen. Ein Protokoll wird bestimmt durch die jeweilig zu unternehmende Kooperationsaufgabe; von besonderer Bedeutung sind dabei die Fälle, in denen Kontrollinstanzen koordinierte Aktionen ausführen müssen, z.B. bei der koordinierten Betriebsmittelzuweisung für einen Auftrag.

Besonders interessant ist der Fall, wenn die Aufträge Operationen auf den Konstituenten der Datenbasis einer Datenbank beinhalten, die über mehrere Kontrollinstanzen verteilt ist.
In diesem Fall - und deshalb wurde für diese Fallstudie als Anwendungsbezug der Datenbankbereich gewählt - existieren Koordinierungsprobleme verschiedenartigster Natur, die sich aus Forderungen an die Datenbankführung ergeben wie z.B.

- die Aufrechterhaltung innerer Beziehungen von Daten,
- die Tolerierung des Ausfalls von Datenmengen oder Kontrollinstanzen,
- die Wiedereingliederung von reparierten Komponenten.

Beschreibt man eine Datenbasis als Menge von Objekten der Art (Name, Wert), wobei der Wert elementar - z.B. eine Zahl - oder selbst wieder ein Objekt sein kann, so umfaßt eine Konstituente eine Menge von Datenbasisobjekten.

Konstituente können

- semantisch identisch (redundante Realisierung von Objekten),
- strukturell abhängig (in einer definierten Beziehung stehend und nicht notwendig semantisch identisch),
- unabhängig sein.

Aufbau und Inhalt der Datenbasis einer Datenbank sind durch ein Schema /D2,L3/ festgelegt, dem alle Daten in der Datenbasis ge-

nügen müssen. Im Schema sind auch die Spezifikationen seitens der Umwelt über die strukturellen Abhängigkeiten von Datenbasisobjekten als Konsistenzregeln /S3/ festgelegt, die z.B. Beziehungen zwischen Werten unterschiedlicher Objekte der Datenbasis betreffen (interne Konsistenz /T2,W1/). Zu fordern ist auch, daß Objekte, die redundant realisiert mehreren Konstituenten angehören, semantisch identische Werte besitzen müssen (externe Konsistenz).

Der Zustand einer Datenbasis heißt konsistent, wenn alle Konsistenzregeln simultan erfüllt sind.

Operationen auf der Datenbasis können deren Inhalt

- nicht verändern: Lesen der Werte von Objekten
- verändern: Ersetzen, Hinzufügen oder Löschen von Objekten bzw. deren Werten.

Da die Ausführung inhaltsverändernder Operationen als Teil von Aufträgen eine gewisse Zeit in Anspruch nimmt, befindet sich die Datenbasis währenddessen temporär in nicht konsistentem Zustand. Es wurde daher /E2/ der Begriff der Transaktion als konsistenzbewahrende Einheit für den Zugriff auf Datenbasisinformation eingeführt:

- Operationen auf der Datenbasis dürfen nur innerhalb einer Transaktion durchgeführt werden (endliche Sequenz von Operationen mit endlicher Ausführungszeit).
- Zu Beginn und Ende einer Transaktion befindet sich die Datenbasis in konsistentem Zustand; während der Ausführung der Operationenfolge der Transaktion kann die Datenbasis in einem inkonsistenten Zustand sein.
- Im Fehlerfall (Ausfall von Datenbasiskomponenten) dient sie als Einheit für die Wiederherstellung eines funktionsfähigen Betriebs.

Assoziiert mit einer Transaktion ist ein Konsistenzbereich als der Menge der Objekte, die in ihrer Gesamtheit in einem für die Transaktion konsistenten Zustand sein muß; der Konsistenzbereich kann Konstituenten mehrerer Kontrollinstanzen umfassen.

Wir setzen im folgenden voraus, daß Transaktionen die Konsistenzregeln kennen und berücksichtigen; ansonsten sind Maßnahmen zur Erhaltung der semantischen Integrität /B1,B2/, d.h. zur Sicherung der Korrektheit und Vollständigkeit der Daten aus der Sicht der Umwelt, vom System bereitzustellen.

Unter operationaler Integrität /B1,B2/ verstehen wir die Erhaltung der Konsistenz der Datenbasis bei paralleler Ausführung unterschiedlicher Transaktionen. Wird die parallele Ausführung von Transaktionen erlaubt, so können Konsistenzverletzungen der Datenbasis eintreten, falls die Konsistenzbereiche von Transaktionen sich überlappen und inhaltsverändernde Operationen nicht sorgfältig synchronisiert werden /E2/; für eine inhaltsverändernde Transaktion müssen daher die Datenbasisobjekte ihres Konsistenzbereichs exklusiv reserviert werden. Ein weiterer Grund für das Sperren von Datenbasisobjekten besteht z.B. darin /E2/, daß Lesevorgänge für Transaktionen reproduzierbar sein müssen. Zur Sicherung der operationalen Integrität müssen Sperrmechanismen existieren, die Konsistenzverletzungen auf Grund von Konflikten parallel ablaufender Transaktionen verhindern.

Lösungen für Sperrmechanismen in zentralisierten Datenbasen wurden in den letzten Jahren entwickelt; sie können in verteilten Datenbasen mit einer zentralen Kontrollstruktur eingesetzt werden.

Für verteilte Datenbasen werden bisher im wesentlichen nur für den Spezialfall redundant realisierter Dateien (Multi-Kopien-Problem) konsistenzsichernde Verfahren für dezentralisierte Kontrolle bereitgestellt; sie berücksichtigen nicht die Kooperation von Kontrollinstanzen, falls Konsistenzbereiche von Transaktionen strukturell abhängige Konstituenten unterschiedlicher Kontrollinstanzen umfassen.

In dieser Arbeit sollen Kommunikationsprotokolle entwickelt werden, die bei einer dezentralen Kontrollstruktur eine parallele Bearbeitung von Transaktionen unter Sicherung der operationalen Integrität der verteilten Datenbasis bei minimalem Koordinierungsaufwand erlauben. Dies geschieht zunächst unter der Annahme eines zuverlässigen Gesamtsystems. Die Kommunikationsprotokolle sollen als Grundlage für den Aufbau dezentraler Kontrollstruktu-

ren zur Transaktionsbearbeitung unter Berücksichtigung von Optimierungskriterien dienen.

Im nächsten Schritt werden die erarbeiteten Kommunikationsprotokolle zu fehlertoleranten Koordinierungsmechanismen erweitert. Derartige Protokolle bilden die unabdingbare Voraussetzung zur Durchführung der bei Ausfall und Wiedereingliederung von Konstituenten und Kontrollinstanzen vom System zu ergreifenden Maßnahmen. Die wesentlichen Maßnahmen sind:

- Erkennung von Fehlern in der Transaktionsbearbeitung,
- Sicherung der operationalen Integrität für alle Transaktionen und Datenbestände des Restsystems über den Störfall hinweg (z.B. Datensicherung, koordiniertes Rücksetzen von Transaktionen),
- Rekonfiguration aller arbeitsfähigen Komponenten zu einem funktionsfähigen Restsystem,
- Aufrechterhaltung der operationalen Integrität bei paralleler Bearbeitung der im Restsystem ablauffähigen Transaktionen,
- Sicherung der Konsistenz für die Wiedereingliederung von Konstituenten.

Schließlich wird im letzten Schritt für die entwickelten Koordinierungsmechanismen die Anwendung neuerer Systemtechnologien zur betriebsgünstigen und aufwandsarmen Realisierung untersucht.

Als wesentliche Randbedingungen für die Konzipierung der Koordinierungsmechanismen sind zu berücksichtigen:

- die Struktur und Topologie der verteilten Datenbasis,
- die Komplexität der auf der Datenbasis zugelassenen Transaktionen (Transaktionsaufbau, Operationsfolgen),
- die Anforderungen unterschiedlicher Sperrmechanismen,
- die Synchronisation der Kontrollinstanzaktivitäten nur durch Nachrichten (nicht etwa durch spezielle Hardware wie Taktsteuerungen),
- die Verzögerung des Nachrichtenaustauschs zwischen Kontrollinstanzen aufgrund der endlichen Übertragungsgeschwindigkeit im Nachrichtentransportsystem.

In Kapitel 2 werden die Anforderungen an Sperrmechanismen zur Sicherung der operationalen Integrität in zuverlässigen Systemen zusammengestellt.

In Kapitel 3 werden existierende Lösungen des Multi-Kopien-Problems hinsichtlich ihrer Eignung als Grundlage für ein allgemein verwendbares Basisprotokoll zur Koordinierung beliebiger Aktivitäten von Kontrollinstanzen geprüft. Ein Basisprotokoll wird selektiert, das zur Koordinierung der Kontrollinstanzen sowohl im Fall eines zuverlässigen Systems als auch im Fehlerfall eingesetzt werden kann.

In Kapitel 4 werden die Kommunikationsprotokolle zur Koordination von Sperrungen bei zuverlässigem Gesamtsystem erarbeitet und in Kapitel 5 für den Fehlerfall erweitert.

Kapitel 6 behandelt die Realisierung von dezentral organisierten Koordinierungsmechanismen.

2. Strategien und Einsatzmöglichkeiten für Sperrmechanismen

Im folgenden werden die von dezentral organisierten Kontrollmechanismen zu berücksichtigenden Anforderungen zur Sicherung der operationalen Integrität bei Zugriffen von parallel ablaufenden Transaktionen auf Datenbasisobjekte zusammengestellt.

2.1. Datenbasisobjekte als Betriebsmittel für Transaktionen

Die exklusive Reservierung von Datenbasisobjekten für Transaktionen kann die Blockierung anderer Transaktionen implizieren und sogar zu gegenseitigen Blockierungen, zu Verklemmungen /K1/, von Transaktionen führen. Verklemmungen können allerdings nur auftreten, falls alle folgenden Bedingungen V1-V5 simultan erfüllt sind /E3/:

V1: Objekte werden für Transaktionen exklusiv reserviert.

V2: Zwei oder mehr Transaktionen dürfen sich gleichzeitig um die exklusive Reservierung von Objekten bewerben.

V3: Eine Transaktion darf die exklusive Reservierung zusätzlicher Objekte beantragen, obwohl sie bereits exklusive Kontrolle über andere Objekte besitzt.

V4: Einer Transaktion dürfen exklusiv reservierte Objekte nicht entzogen werden, bevor sie nicht diese selbst freigibt.

V5: Es existiert eine zyklische Kette von Transaktionen derart, daß jede Transaktion exklusive Kontrolle über Objekte besitzt, um die sich ihr Nachfolger in der Kette bewirbt.

Im Bereich der Betriebssystemforschung wurden unterschiedliche Strategien zur Behandlung des Blockierungsproblems entwickelt, die sich in die Klassen K1 und K2 einordnen lassen /C5,E3/:

K1: Verhinderung von Verklemmungen
Diesen Verfahren liegt die Aufgabe einer der Bedingungen V2-V5 zugrunde:

- Aufgabe von V2: Transaktionen, die sich potentiell um exklusive Reservierung derselben Objekte bewerben, werden durch eine externe Instanz in einer Reihenfolge angeordnet, in der sie seriell ablaufen dürfen.

- Aufgabe von V3: Eine Transaktion darf erst ihre Ausführung beginnen, falls alle exklusiv benötigten Objekte für sie entsprechend reserviert sind; während ihrer Ausführung darf sie sich nicht um die exklusive Reservierung zusätzlicher Objekte bewerben (hierzu existieren mehrere Varianten /C5/, die mehr Parallelität von Transaktionen gestatten).
- Aufgabe von V4: Einer Transaktion werden alle bereits exklusiv reservierten Objekte entzogen, falls sie eine Verklemmung hervorrufen könnte.
- Aufgabe von V5: Alle Objekte, die nicht entzogen werden dürfen und möglicherweise exklusiv benutzt werden könnten, werden linear angeordnet. Nur in der durch die Anordnung vorgegebenen Reihenfolge dürfen die Objekte belegt werden.

K2: Erkennung und Beseitigung von Verklemmungen
Bei diesen Verfahren werden Verklemmungen zwischen Transaktionen zugelassen. Auf Anforderung durch eine Transaktion (falls eine Transaktion auf die Freigabe von Objekten durch eine oder mehrere Transaktionen warten muß) oder in periodischen Zeitabständen wird untersucht, ob eine gegenseitige Blockierung aufgetreten ist.

Ein mögliches Verfahren zur Verklemmungserkennung besteht darin, die Wartebeziehungen zwischen den Transaktionen in einem gerichteten Graph, dem Blockierungsgraph, zu erfassen und auf Zyklen zu untersuchen /B1,K1,S2/.

Der Blockierungsgraph - siehe Bild 2.1 - besitzt als Knotenmenge die Menge der im System befindlichen Transaktionen $\{T_1,\ldots,T_n\}$. Eine gerichtete Kante von T_i nach T_j, $i,j=1(1)n$, $i\neq j$, existiert genau dann, wenn T_i auf die Freigabe von Objekten, die für T_j exklusiv reserviert sind, wartet, d.h. wenn T_i durch T_j blokkiert ist. Die Kante wird mit den freizugebenden Objekten markiert. Da bei exklusiver Reservierung von redundant realisierten Datenbasisobjekten alle Exemplare exklusiv reserviert sein müssen, ist für die Existenz einer Verklemmung das Vorhandensein eines Zyklus von Transaktionen im Blockierungsgraph notwendig und hinreichend.

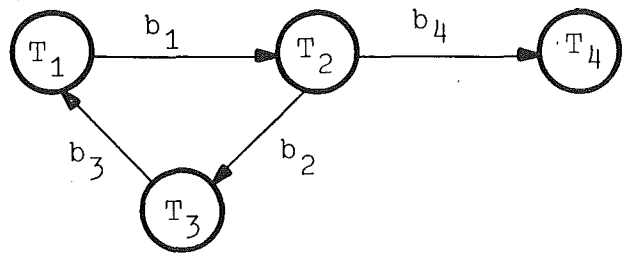


Bild 2.1: Beispiel für einen Blockierungsgraph für die Menge der Transaktionen $\{T_1,T_2,T_3,T_4\}$ und die Menge von Objekten $\{b_1,b_2,b_3,b_4\}$. Der Zyklus T_1,T_2,T_3,T_1 zeigt eine Verklemmungssituation an.

Die Beseitigung von Verklemmungen erfordert den Entzug von exklusiv reservierten Objekten von Transaktionen des Zyklus. Diese Transaktionen werden in Abhängigkeit von z.B. der Höhe der Entzugskosten oder der Anzahl von Zyklen, an denen sie beteiligt sind, bestimmt.

Welche Strategien zur Behandlung des Blockierungsproblems herangezogen werden, hängt wesentlich von der Struktur der Datenbasis ab und wird von der Natur des Betriebsmittels "Datenbasisobjekt" mitbestimmt.

Solche Charakteristika sind /C2/:

- Objekte können von einer Transaktion nicht nur über ihre Benennung, sondern auch über ihren Inhalt bestimmt werden; dies erschwert die Unterscheidung der Sperrbereiche unterschiedlicher Transaktionen und die Anwendung der Strategie K1 mit Aufgabe von V3.
- Transaktionen können die Eigenschaften von Objekten ändern.
- Eine Transaktion möchte Objekte sperren und aufgrund ihres Inhalts untersuchen, ob weitere Objekte gesperrt werden müssen. Dies bedeutet, daß Strategie K1 mit Aufgabe von V5 ausgeschlossen ist.
- Um einen hohen Parallelitätsgrad von Transaktionen zu gestatten, sollte die Granularität von Sperrungen sehr fein sein /G2/. Hierdurch entstehen Größenordnungen von Betriebsmitteln, was den Einsatz von Strategien beider Klassen K1 und K2 erschwert.

- Transaktionen können Objekte eliminieren.
- Transaktionen können neue Objekte in die Datenbasis einbringen. Dies kann die Erzeugung von "Phantomen" /B1,E2/ implizieren. Phantome können zu Konsistenzverletzungen der Datenbasis führen, denn durch sie können korrekt ablaufende Transaktionen, die zu Beginn ihrer Ausführung einen konsistenten Datenbasiszustand vorfanden und die benötigten Objekte sperrten, nach ihrer Ausführung einen inkonsistenten Zustand der Datenbasis hinterlassen (Integrität von Transaktionen /T1/).

  Die Sperrung physisch existierender Objekte - physische Sperrung - ist zur Konsistenzhaltung der Datenbasis daher nicht ausreichend. Die Materialisation potentieller Objekte als Phantome kann nur durch eine logische Sperrung (Prädikatssperrung) verhindert werden /B1,E2/.

Aufgrund dieser Eigenheiten des Betriebsmittels "Datenbasisobjekt" werden bei zentralisierten Datenbasen der Parallelität von Transaktionen oft enge Grenzen gesetzt, etwa durch Reservierung von bestimmten Betriebszeiten, innerhalb der nur Änderungen in die Datenbasis (i.a. seriell) eingearbeitet werden dürfen und reine Leser-Transaktionen nicht zugelassen sind, oder es werden Verfahren verwendet, die eine Verklemmung zulassen und die weniger auf Anforderung, sondern bevorzugt periodisch /G3/ beseitigt werden.

Um den Aufwand für den Entzug von bereits reservierten Objekten möglichst gering zu halten, da u.U. umfangreiche Veränderungen der Datenbasis rückgängig zu machen wären, wird z.B. vorgeschlagen,

- Änderungen nicht auf der Datenbasis selbst, sondern auf Kopien der entsprechenden Daten durchzuführen und die Änderungen erst bei Transaktionsende in die Datenbasis einzubringen /C2,K1/,
- alle von einer Transaktion für ihre Ausführung benötigten Objekte während einer sog. Bindungsphase zu sperren und danach die eigentliche Ausführung der Operationen der Transaktion zu beginnen. In dieser Bindungsphase können bereits erworbene Objekte im Verklemmungsfall einer Transaktion entzogen werden; dieser Entzug ist, da noch keine Operationen auf Datenbasis-

objekten ausgeführt worden sind, mit relativ geringem Aufwand durchführbar und gefährdet nicht die Konsistenz der Datenbasis.
Um unendliches Warten einer Transaktion durch ständige Verdrängung durch andere Transaktionen zu vermeiden, sind Strategien für die Auswahl zu verdrängender Transaktionen mit der Anzahl ihrer bisherigen Verdrängungen zu koppeln /C2/.

2.2. Physische und logische Sperrung

Die zu berücksichtigenden Einsatzmöglichkeiten von Sperrmechanismen in verteilten Systemen müssen sich an den folgenden Sperrungsarten orientieren:

- physische Sperrung von Objekten nach Benennung: diese Sperrung bezieht nur existierende Objekte ein,
- logische Sperrung von Objekten: diese umfaßt die Sperrung von existierenden und potentiellen Objekten.

Zu beiden Sperrungsarten existieren für zentralisierte Datenbasen unterschiedliche Lösungsvorschläge, deren Effizienz von der Struktur der Datenbasis i.a. wesentlich beeinflußt wird.

Die Grundlage für Sperrmechanismen ist die Einteilung der Datenbasis in Sperreinheiten, die als Ganzes für Transaktionen gesperrt werden und die damit für die Sperrmechanismen die an die Transaktionen zu vergebenden Betriebsmittel sind. Über die Sperreinheiten ist Sperrinformation zu führen, die deren Sperrungszustand reflektiert und hierzu Angaben enthält über

- den Verfügbarkeitszustand einer jeden Sperreinheit,
- die Blockierungssituation von Transaktionen (z.B. Blockierungsgraph), die aus den Belegungen der Warteschlangen der Sperreinheiten gewonnen werden kann.

Mit einer Sperranweisung spezifiziert eine Transaktion gegenüber dem Sperrmechanismus die vorzunehmende Sperrung unter Angabe

- der zu sperrenden Objekte bzw. Objektmengen,
- der Sperrungsstufe zu jedem der zu sperrenden Objekte bzw. Objektmengen.

Die Einteilung der Datenbasis in Sperreinheiten ist einer Transaktion i.a. nicht bekannt; sie sollte diese Kenntnis auch nicht besitzen müssen, schon um sie vom angewandten Sperrmechanismus oder dem momentanen Zustand einer dynamisch veränderbaren Verteilungsstruktur der Datenbasis unabhängig zu halten. Die Transformation zwischen der Sperranweisung einer Transaktion und der systeminternen Sperranweisung ist daher durch das System vorzunehmen. Aus Vereinfachungsgründen unterscheiden wir im folgenden nicht zwischen den Sperrobjekten von Transaktionen und den systeminternen Sperreinheiten.

Die anzugebenden Sperrungsstufen hängen von der Granulierung der Sperrungen ab, die vom System erlaubt sind. Die Granulierung von Sperrungen kann einheitlich sein, z.B. könnten als Sperreinheiten im Relationenmodell /C4/ entweder nur Relationen oder nur Tupel auftreten. Bei unterschiedlicher Sperrungsgranulierung /G2/ existiert z.B. eine Hierarchie von Sperreinheiten (siehe Bild 2.2) und den Transaktionen ist es gestattet, auf unterschiedlichen Ebenen der Hierarchie (dargestellt als gewurzelter Baum) zu sperren.

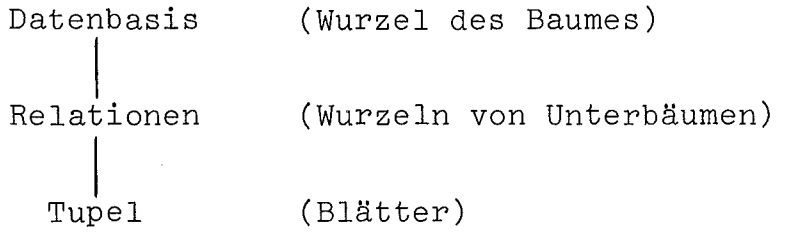


Bild 2.2: Beispiel für eine Hierarchie von Sperreinheiten

Bei physischer Sperrung von Objekten nach Benennung werden die zu sperrenden Objekte bzw. Objektmengen über ihren Namen angegeben.

Die Sperrungsstufe ist bei Sperreinheiten einheitlicher Granulierung von der Art

- Keine Sperrung, falls das Objekt verfügbar ist,
- Lesesperrung, falls eine Transaktion die Sperreinheit während ihrer Ausführungsphase nur lesen aber nicht verändern will,

- Veränderungssperrung, falls eine Transaktion die Sperreinheit modifizieren will und ihr deshalb exklusiver Zugriff auf sie reserviert werden muß.

Während Lesesperrungen unterschiedlicher Transaktionen bzgl. derselben Sperreinheit gleichzeitig zugelassen werden können, da sie sich nicht stören (Parallelität von Lesern), kollidieren Lesesperrungen mit Veränderungssperrungen und Veränderungssperrungen untereinander.

Für die Sperrungsstufe einer Sperreinheit ist in der Sperrinformation festzuhalten, ob sie

- verfügbar,
- mit Lesesperrung belegt,
- mit Veränderungssperrung belegt

ist; im Falle einer Lesesperrung ist die Anzahl der momentan lesenden Transaktionen mitzuführen.

In Datenbasen mit Sperreinheiten unterschiedlicher Granulierung sind mehrere Arten von Sperrungsstufen mit differenzierten Kollisionsbeziehungen notwendig /G2/, um Sperrungen auf unterschiedlichen Granulierungsebenen zu erlauben. Ausgehend von der Ebene der gröbsten Granularität - in Bild 2.1 die Datenbasis - muß auf jeder Ebene durch die entsprechende Sperrungsstufe angezeigt werden, ob die eigentliche Sperrung erst auf einer Ebene feinerer Granulierung vollzogen werden soll oder ob die momentan erreichte Ebene mit allen ihren Nachfolgern als gesperrt zu betrachten sind. Freigaben erfolgen von den Blättern aus in Richtung Wurzel. In /G3/ werden z.B. 6 Sperrungsstufen unterschieden.

Mehrere Transaktionen unterschiedlicher Sperrungsstufe bzgl. eines Knotens können gleichzeitig zugelassen werden, falls ihre Sperrungsstufen mit der Sperrungsstufe des Knotens kompatibel sind; die Sperrungsstufe des Knotens richtet sich nach dem Supremum der Sperrungsstufen der momentan zugelassenen Transaktionen.

In /G3/ sind Erweiterungen aufgezeigt, die über die (in Bild 2.2 dargestellten) baumartigen Zuordnungen von Sperreinheiten hinaus azyklische Strukturen gestatten.

Bei der physischen Sperrung von Objekten nach Benennung kann die Transaktion auch die zu sperrenden Objekte über ein Prädikat festlegen /C2/ z.B. alle Tupel einer Relation, deren Attribut A einen bestimmten Wert besitzen. In diesem Fall muß eine Transaktion erst die entsprechenden Benennungen der zu belegenden Sperreinheiten ermitteln. Dies erfolgt während eines Suchvorgangs auf der Datenbasis, der durch eine der Transaktion zugeteilten "Suchmaschine" durchgeführt wird. Die Transaktion befindet sich in dieser Zeit in der Bindungsphase /C2/. Kann eine Belegung der Sperreinheit durchgeführt werden, wird die Sperrungsstufe bei der Sperreinheit in der Sperrinformation entsprechend vermerkt, ansonsten wird die Transaktion nur in die der Sperreinheit zugehörenden Warteschlange eingereiht.

Kann die Bindungsphase erfolgreich beendet werden, so tritt die Transaktion in ihre Ausführungsphase ein, in der ihre Operationsfolge abgearbeitet wird. Bei Beendigung der Transaktion werden die reservierten Betriebsmittel alle freigegeben. Hat eine Transaktion in ihrer Ausführungsphase Datenbasisteile modifiziert, so müssen die durch sie blockierten Transaktionen anschließend prüfen, ob durch die Datenbasisänderung ihre Anforderung auf Belegung des Objektes nicht etwa hinfällig geworden ist.

Bei der logischen Sperrung /B1,E2/ hat die Transaktion eine Prädikatssperrung anzugeben, die sowohl existierende als auch potentielle Datenbasisobjekte sperrt. Eine solche Prädikatssperrung kann z.B. für eine n-stellige Relation wie folgt spezifiziert werden /E2/:

(Name der Relation, Prädikat, {(Attributname i, Sperrungsstufe):
i = 1(1)n})

Eine solche Prädikatssperrung wird als Sperreinheit aufgefaßt, d.h. die Sperrinformation umfaßt alle genehmigten Prädikatssperrungen zusammen mit den Urheber-Transaktionen. Die Prädikatssperrung einer neu hinzukommenden Transaktion wird mit den bereits existierenden verglichen /E2/ (zur Problematik des Vergleichs von Prädikaten siehe /K3/); im Konfliktfall wird die Transaktion blockiert und in die Warteschlange für die entsprechende Prädikatssperrung eingereiht.

Für den Spezialfall, daß im Transaktionsprofil reine Leser unterschieden werden können, empfiehlt /B1/ aus Implementierungsgründen eine Kombination zwischen logischer und physischer Sperrung. Leser werden nur einer physischen Sperrung unterzogen, die anderen Transaktionen müssen vor der physischen Sperrung die logische Sperrung erfolgreich durchführen. Die physische Sperrung durch die Transaktionen erfolgt in einer Bindungsphase wie bei der physischen Sperrung, falls die Sperrobjekte durch ein Prädikat festgelegt sind /B1/.

2.3. Phasen der Transaktionsbearbeitung

Als hinreichend für die Erhaltung der Konsistenz der Datenbasis (und der Integrität von Transaktionen) werden in /T1/ (auf der Basis von /E2/) folgende Bedingungen genannt:

I1. Jede Transaktion hält Datenbasisobjekte bei Benutzung entsprechend gesperrt, d.h. bei Lesezugriffen mit Lesesperrung oder Veränderungssperrung, bei verändernden Zugriffen mit Veränderungssperrung.

I2. Eine Transaktion darf nach der ersten Freigabe von Objekten keine weiteren Sperrungen fordern. Dies impliziert eine zweiphasige Grobstruktur der Transaktion in
- Wachstumsphase: innerhalb dieser darf die Transaktion Objekte zur Sperrung anfordern,
- Schrumpfungsphase: sie beginnt mit der ersten Freigabe von Objekten; weitere Sperrungen dürfen nicht mehr angefordert werden.

I3. Eine Transaktion muß bei Beendigung alle von ihr noch gesperrten Objekte freigeben.

I4. Kein Objekt der Datenbasis ist gleichzeitig mit kollidierenden Sperrungsstufen belegt.

I5. Für jede Transaktion gilt während ihrer gesamten Laufzeit, daß sie durch ihre Sperrung von Objekten gegen Phantome geschützt ist.

Die angegebenen Bedingungen sind i.a. für die Konsistenz der Datenbasis nur hinreichend aber nicht notwendig /T1/, da für Spezialfälle von Datenbasen weniger einschränkende Bedingungen

für die Integritätssicherung ausreichen.

Anhand der Bedingungen I1, I2 und I3 ergibt sich das in Bild 2.3 dargestellte Phasendiagramm für eine Transaktion.

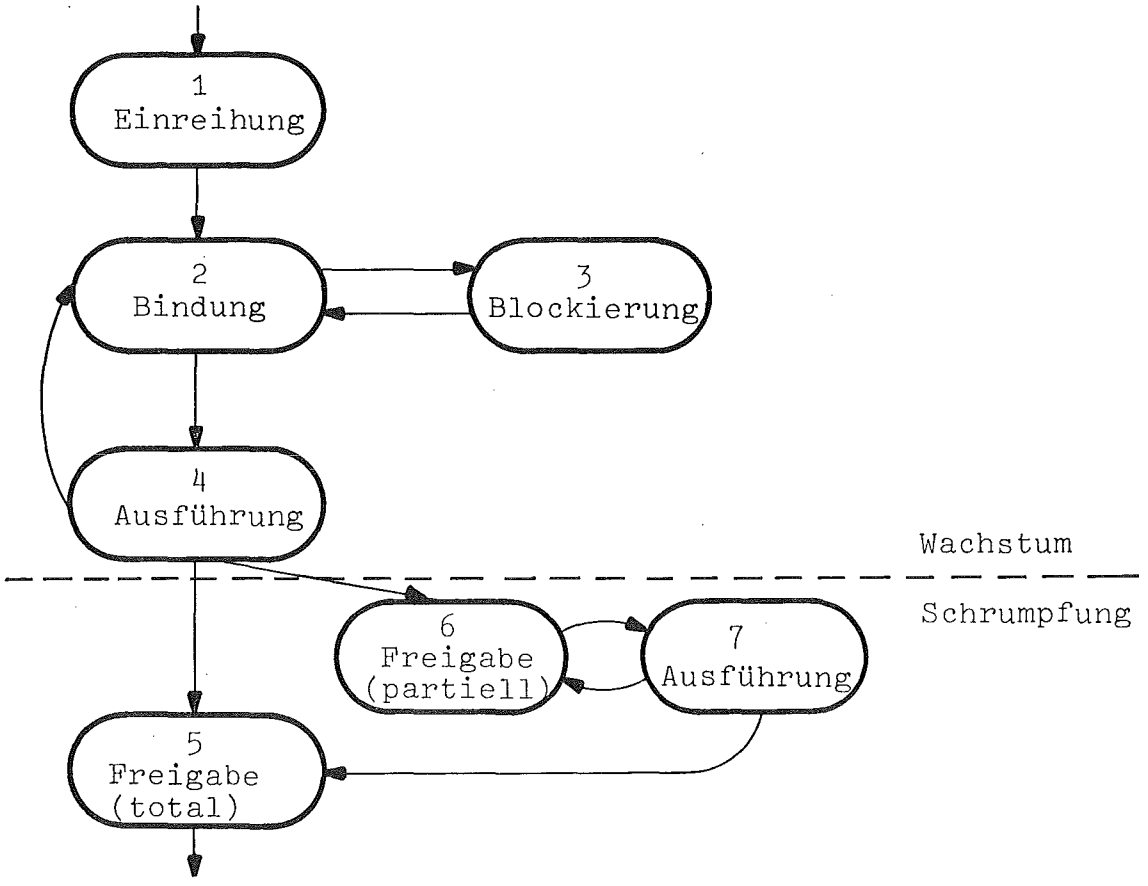


Bild 2.3: Phasendiagramm einer Transaktion

Phase 1 beinhaltet die Aufnahme der Transaktion in die Menge der momentan zur Bearbeitung zugelassenen Transaktionen und ist u.U. mit einer prioritätsgesteuerten Einreihung in diese verbunden.
Danach müssen die von der Ausführung von Transaktionsoperationen betroffenen Objekte gesperrt werden (Bedingung I1). Diese Sperrung erfolgt in der Phase 2, der Bindungsphase.

Können wegen einer Kollision mit existierenden Sperrungen von der Transaktion benötigte Objekte nicht mit der erforderlichen Sperrungsstufe für sie reserviert werden, wird die Transaktion blockiert und muß auf die Freigabe der Betriebsmittel warten. Verfolgt der Sperrmechanismus die Strategie der Verklemmungsbeseitigung, kann die Blockierung beendigt werden durch

- Zuteilung der benötigten Betriebsmittel durch Verdrängung der sie reserviert haltenden Transaktionen und Erlaubnis zur Weiterführung der Bindungsphase,
- Annullierung aller bisher erworbenen Anrechte, d.h. Entzug aller bisher reservierten Betriebsmittel, und Neubeginn der Bindungsphase.

Bei der Strategie der Verklemmungsverhinderung besteht die Bindungsphase aus einer unteilbaren Aktion, während der versucht wird, alle benötigten Objekte auf einmal zu reservieren; dies kann nur angewendet werden, falls die zu sperrenden Objekte bekannt sind und nicht durch Zugriff (z.B. durch "Suchmaschinen" /C2/) auf die Datenbasis erst ermittelt werden müssen.

Nach erfolgreichem Abschluß der Bindungsphase darf die Transaktion in der Ausführungsphase die gewünschten Manipulationen auf den von ihr gesperrten (existierenden, potentiellen) Objekten vornehmen. Sollten aufgrund von Ergebnissen, die während der Ausführungsphase anfallen, zusätzliche Objekte gesperrt werden, ist eine weitere Bindungsphase einzuleiten. Kommt es bei dem Versuch, zusätzliche Betriebsmittel zu sperren, zu Blockierungen oder Verklemmungen, so können zur weiteren Aufrechterhaltung der Konsistenz der Datenbasis nicht nur die bereits zusätzlich reservierten, sondern auch die während früherer Bindungsphasen belegten Objekte entzogen werden. Dies bedeutet eine totale Rücksetzung der Transaktion, bei der alle von der Transaktion eingebrachten Änderungen der Datenbasis rückgängig gemacht werden müssen.

Werden nicht alle Betriebsmittel auf einmal freigegeben (partielle Freigabe), kann eine weitere Ausführungsphase folgen. Diese darf jedoch wegen Bedingung I2 nicht die Phase 4 sein, in der zusätzliche Sperrungen beantragt werden dürfen, sondern muß als eigenständige Phase (Phase 7) geführt werden.

Die Freigabe aller noch belegten Objekte erfolgt in der Phase 5, mit deren Abschluß die Transaktion als beendet gilt.

2.4. Annahmen für die weitere Problembearbeitung

Das Gebiet der Datenbanktechnologie dient in dieser Arbeit als Anwendungsbezug zur Fallstudie über dezentral organisierte Betriebsmittelverwaltung. Es steht deshalb nicht die Lösung aller mit der Führung von Datenbasen zusammenhängenden Probleme im Vordergrund.

Für die weitere Problembearbeitung werden daher folgende Annahmen über Transaktionen und Datenbasen vorausgesetzt:

- Transaktionen bestehen nur aus den Phasen 1-5, d.h. partielle Freigaben von gesperrten Objekten sollen nicht erlaubt sein. Diese Forderung stellt keine wesentliche Einschränkung dar, da aus Gründen, die mit Maßnahmen zur Wiederherstellung einer funktionsfähigen Datenbasis nach Störsituationen zusammenhängen, gesperrte Objekte erst am Ende einer Transaktion freigegeben werden dürfen /B4/.
- Änderungen werden nicht auf der Datenbasis selbst, sondern auf Kopien der entsprechenden Daten durchgeführt, um die Transaktionsrücksetzung zu erleichtern (vgl. 2.1.).
- Die Datenbasis ist in Sperreinheiten einheitlicher Granularität eingeteilt.
- Transaktionen spezifizieren die von ihnen benötigten Datenbasisobjekte über deren Benennung und nicht über deren Inhalt.

Die Aufhebung der beiden letztgenannten Einschränkungen wird in 4.6. diskutiert.

3. Ein Basisprotokoll zur dezentralisierten Koordination

Das Hauptproblem der dezentralen Kontrollstruktur ist die Koordinierung von Aktivitäten der Kontrollinstanzen zur Transaktionsbearbeitung und zur Reaktion im Fehlerfall. Lösungsansätze für Verfahren mit dezentraler Kontrollstruktur wurden im wesentlichen nur für die Multi-Kopien-Haltung in verteilten DV-Systemen entwickelt. Wichtige Verfahren sollen im folgenden diskutiert und auf ihre Eignung als Basis für Kommunikationsprotokolle zur Behandlung allgemeinerer Koordinationsaufgaben geprüft werden /H7/. Ein Basisprotokoll, das als Ausgangspunkt für Verfahren zur Koordinierung der Kontrollinstanzen sowohl für die Transaktionsbearbeitung (siehe Kap. 4) als auch für die Reaktion im Fehlerfall (siehe Kap. 5) Anwendung finden kann, wird im Detail vorgestellt und seine Korrektheit untersucht.

3.1. Existierende Verfahren zur Behandlung des Multi-Kopien-Problems

Unterschiedliche Verfahren ergeben sich aus

- der Art der gewählten Kontrollstruktur: zentral oder dezentral,
- dem verfolgten Prinzip zur Einbringung von Änderungen in die Kopien der Datenbasis.

Gemeinsame Voraussetzung aller folgenden Verfahren zur Kopienführung ist eine eineindeutige Zuordnung zwischen Kopien und Kontrollinstanzen.

3.1.1. Verfahren mit zentraler Kontrollstruktur

In /A2/ wird ein Verfahren entwickelt, das die interessante Eigenschaft der "2-Rechner-Resilienz" (2-host-resiliency) besitzt: nur ein gleichzeitiger Ausfall von 2 Kontrollinstanzen während einer kritischen Phase in der Bearbeitung eines Benutzerauftrags (Datenbasisveränderung) impliziert einen Abbruch der Auftragsausführung.

Über die Datenbasisstruktur werden keine Einschränkungen vorausgesetzt. Die Kontrollinstanzen werden einer linearen Rangordnung unterworfen und als Zentrale die Kontrollinstanz mit dem höchsten Rang eingesetzt.

Der Benutzer - siehe Bild 3.1 - übergibt die Anforderung auf Datenbasisveränderung an die Zentrale (oder einer anderen Kontrollinstanz, die die Anforderung an die Zentrale weiterleitet).

Die Zentrale führt die Änderung ihrer Kopie aus und übergibt die Benutzeranforderung in einer Nachricht des Typs A an die in der Rangordnung unmittelbar folgende Kontrollinstanz. Diese führt ebenfalls die Änderung durch und sendet Bestätigungen (B_1) an die Zentrale und an den Urheber der Änderung, den Benutzer.

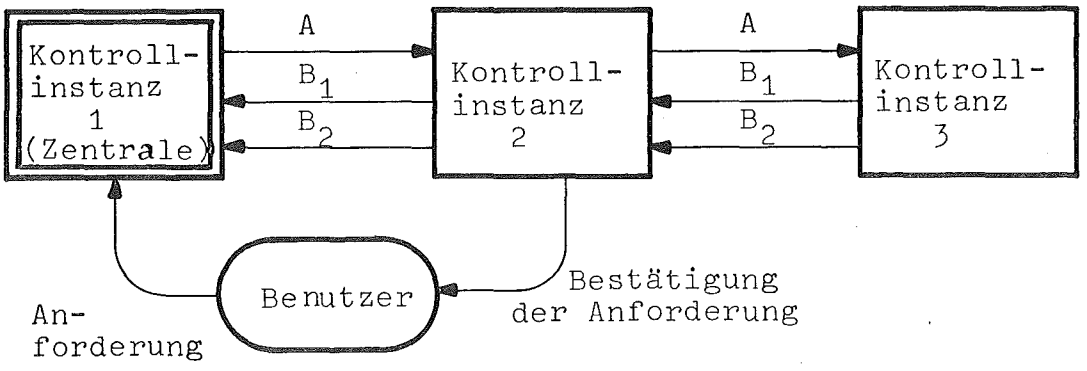


Bild 3.1: Zentrale Kontrollstruktur für "2-Rechner-Resilienz" (Rangordnung: Rang(1) > Rang(2) > Rang(3))

Die restlichen Kontrollinstanzen erhalten die Änderungs-Anforderung entsprechend ihrer Position in der Rangordnung und müssen sie durch Nachricht B_1 ihrem unmittelbaren Vorgänger quittieren. Zur Gewährleistung der 2-Rechner-Resilienz wird zusätzlich von einer Kontrollinstanz eine Bestätigung B_2 an die nächsthöhere Kontrollinstanz gesendet, sobald sie von ihrem direkten Nachfolger die Bestätigung B_1 vorliegen hat. Das Protokoll kann im Fehlerfall nicht die 2-Rechner-Resilienz gewährleisten, falls eine Kontrollinstanz sich fälschlicherweise selbst zur Zentrale erklärt /A2/. Diese Tatsache unterstützt die Annahme, daß eine dezentrale Kontrollstruktur zur Behandlung von Fehlerfällen leistungsfähiger als eine zentrale Kontrollstruktur ist.

3.1.2. Verfahren mit dezentraler Kontrollstruktur

3.1.2.1. Sortierverfahren

Diese Verfahren lassen sich abstrakt beschreiben als Sortieren von Operationen auf der Datenbasis unter Berücksichtigung eines Systemfortschrittsmaßes /F1/. Das Systemfortschrittsmaß (z.B. die Zeit oder ein Zähler) ist eine globale Größe, die entweder an einer Stelle geführt und für alle Kontrollinstanzen zugreifbar sein muß (gemeinsame Uhr) oder verteilt geführt und oft synchronisiert werden muß (jede Kontrollinstanz benutzt ihre eigene (lokale) Uhr).

Dazu werden folgende Ansätze vorgeschlagen:

Verfahren von Johnson /J1,F1/:

Jedes Objekt der Datenbasis ist ein 5-Tupel (N,W,L,G,Z) mit der Bedeutung:

- N ist der Name des Objekts und W sein gegenwärtiger Wert,
- L markiert den Zustand gelöscht/nicht gelöscht,
- G enthält den Zeitstempel der Objektgenerierung,
- Z enthält den Zeitstempel der Modifikation, die für den gegenwärtigen Inhalt von W und/oder L verantwortlich ist.

Zeitstempel setzen sich aus einer Zeitangabe und der Identifikation der für die Zeitangabe verantwortlichen Kontrollinstanz zusammen und dienen gleichzeitig als Prioritätsmerkmal.

Auf der Datenbasis sind folgende Operationen erlaubt: Lesen, Kreieren und Löschen (Entfernen) von Objekten sowie die Zuweisung von Werten an Objekte. Eine Operation ist mit ihrem Namen und dem von ihr betroffenen Objekt anzugeben.

Erhält eine Kontrollinstanz - von lokaler Seite (Eigeninitialisierung) oder von einer anderen Kontrollinstanz (Fremdinitialisierung) - eine Anforderung zur Modifikation der Datenbasis, so wird diese akzeptiert und in die Kopie eingearbeitet, falls ihr Zeitstempel aktueller ist als der Zeitstempel des entsprechenden Objekts, ansonsten verworfen.

Ein Objekt darf durch die Kontrollinstanz erst aus der Kopie der Datenbasis entfernt werden, falls sein Zeitstempel Z älter ist als das Minimum des Vektors LAST-SYNCHRONIZED, den jede

Kontrollinstanz neben dem Vektor LAST-HEARD-FROM zu führen hat. LAST-HEARD-FROM enthält die Zeitstempel Z der letzten Botschaft, die sie von jeder anderen Kontrollinstanz empfangen hat. Bei Eingang einer Lösch-Anforderung übermittelt die Kontrollinstanz allen anderen in einer Botschaft mit Zeitstempel das Minimum ihres Vektors LAST-HEARD-FROM, das diese in ihren Vektor LAST-SYNCHRONIZED an entsprechender Stelle eintragen.

Verfahren von Thomas /T2/:

Die Datenbasis besteht aus einer Menge von Variablen, für die neben ihrem Wert auch der Zeitstempel ihrer letzten Wertänderung geführt wird. Erlaubte Operationen sind Lesen oder Ändern von Objektwerten.

Eine Anforderung auf Wertänderung muß enthalten:

- die neuen Werte der zu ändernden Variablen (U-Variablen),
- die Liste der Variablen, auf denen die Änderungsberechnung basiert (B-Variablen), sowie deren Zeitstempel (Forderung: die B-Variablen müssen zur Konsistenzsicherung die U-Variablen mit einschließen).

Die Kontrollinstanz, der diese Anforderung von lokaler Seite übergeben wird, gibt ihr einen Zeitstempel T mit

T = max (lokale Uhrzeit, 1 + max (Zeitstempel der B-Variablen)).

Diese Zeitstempelfestlegung verhindert zusammen mit einer Prioritätsregelung für Kontrollinstanzen Sequenzanomalien, die durch asynchron arbeitende lokale Uhren verursacht werden könnten.

Das Verfahren versucht, die Anzahl der zu einer für alle verbindlichen Mehrheitsentscheidung notwendigen Abstimmungsschritte zu minimieren. Hierzu werden die Anforderungen auf eine "Abstimmungsreise" geschickt. Jede Kontrollinstanz, bei der eine Anforderung eintrifft, entscheidet anhand der Variablen und der Zeitstempel, ob die Anforderung akzeptiert, abgelehnt oder verzögert behandelt werden muß. Nach erfolgter Entscheidung, die einer Kontrollinstanz nur einmal pro Anforderung erlaubt ist, prüft sie aufgrund der bei der Anforderung mitgeführten bisherigen Entscheidungsergebnisse, ob ein "Konsens" für Ablehnung oder Annahme existiert. Die anderen Kontrollinstanzen sind

zu benachrichtigen, sobald ein Konsens feststeht (jede Kontrollinstanz muß diese Anforderung in ihre Kopie einarbeiten). Kommt ein Konsens nicht zustande, so ist die Anforderung abzulehnen.

3.1.2.2. Verfahren mit exklusiver Sperrung

Diesen Verfahren /H6,M5/ liegt eine feste Kopplung von Aktionsfolgen der kooperierenden Kontrollinstanzen zugrunde. Eine Operation auf der Datenbasis wird erst zugelassen, wenn die betroffenen Objekte für sie exklusiv gesperrt worden sind. Sperrung und Freigabe von Objekten werden nur nach gemeinsamer Absprache der Kontrollinstanzen vollzogen.

Verfahren von Holler /H6/:

Die Struktur der Datenbasis unterliegt keinen Einschränkungen. Als Sperreinheit wird die gesamte Datenbasis betrachtet. Die Grundzüge des Verfahrens in einer Kontrollinstanz zeigt Bild 3.2.

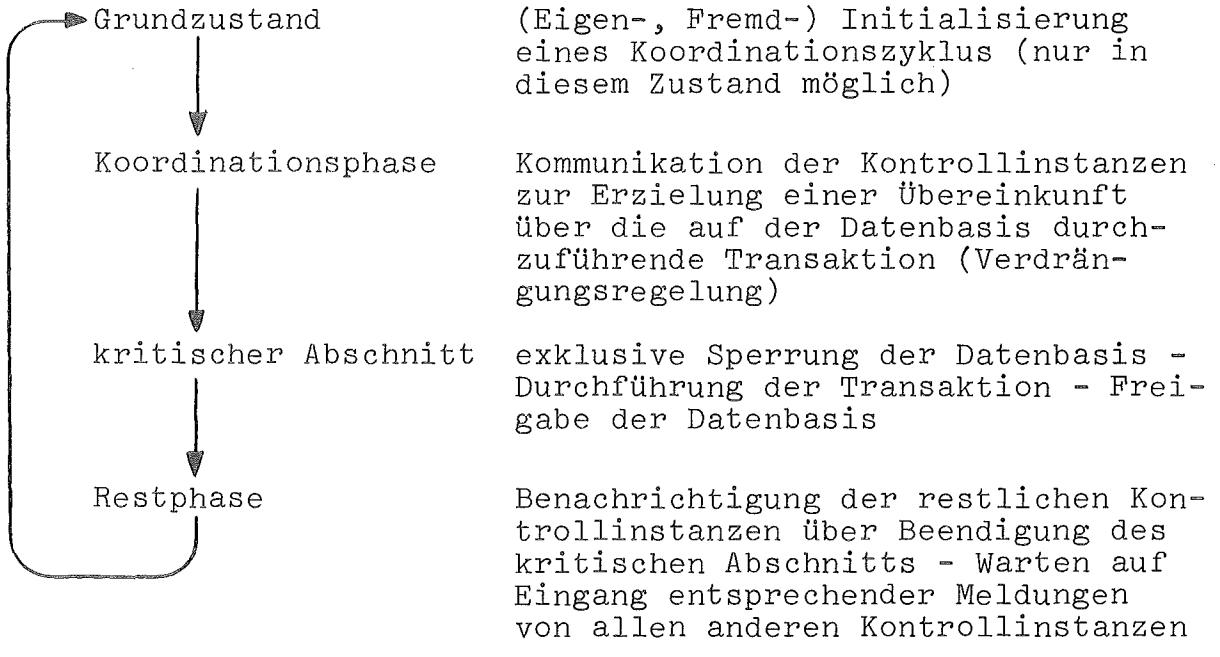


Bild 3.2: Grundzüge des Verfahrens von Holler

Nach Initialisierung eines Koordinationszyklus wartet eine Kontrollinstanz in der Koordinationsphase auf die Bereitschaftserklärungen der restlichen Kontrollinstanzen, die Operation auszuführen, zu deren Durchführung sie sich gegenüber den anderen

Kontrollinstanzen ihrerseits bereit erklärt hat. Nach Erhalt aller Erklärungen geht sie in den kritischen Abschnitt über.

Operationen werden mit eindeutigen Prioritäten (wie bei dem Verfahren von Johnson) versehen. Diese dienen zur Konfliktregelung, falls mehrere Kontrollinstanzen simultan eine Eigeninitialisierung vornehmen. Eine Verdrängung von Operationen ist nur im Grundzustand und in der Koordinationsphase erlaubt. Eine Kontrollinstanz kann einer fremdinitialisierten Verdrängung nur zustimmen, falls sie noch nicht von allen Kontrollinstanzen Bereitschaftserklärungen für eine Operation vorliegen hat. Die Bereitschaftserklärung für eine Alternativ-Operation muß allen Kontrollinstanzen übermittelt werden; entsprechende Bereitschaftserklärungen werden als Bestätigungen zurückerwartet. Eine Kontrollinstanz kann eine Verdrängung nur initialisieren, wenn sie ihrerseits noch keine Bereitschaftserklärungen für eine andere Operation an die restlichen Kontrollinstanzen abgesendet hat und selbst nicht Initiator eines Koordinationszyklus war.

Dieses Protokoll /H6/ sorgt für den Übergang aller Kontrollinstanzen in den kritischen Abschnitt und verhindert, daß diese, falls im kritischen Abschnitt befindlich, unterschiedliche Operationen gestartet haben. Da zudem die geordnete Rückkehr in den Grundzustand gesichert ist, gewährleistet das Protokoll die verklemmungsfreie Koordination von Datenbasisänderungen.

Zur Verhinderung von Sequenzanomalien wird in /H6/ vorgeschlagen, bei erfolgreichem Abschluß der Koordinationsphase nicht sofort in den kritischen Abschnitt überzugehen, sondern erst ein Karenzzeitintervall abzuwarten. Am Ende dieses Intervalls unterrichten sich die Kontrollinstanzen, ob der Übergang in den kritischen Abschnitt vollzogen werden kann oder zuvor eine Verdrängung durchzuführen ist.

Eine Erweiterung des Verfahrens von Holler, die zwischen Lese- und Schreib-Operationen unterscheidet, wird in /M4/ erläutert.

3.1.2.3. Vergleich der Leistungsfähigkeit der Verfahren mit dezentraler Kontrollstruktur

Es sei ein zuverlässiges Gesamtsystem mit n Kontrollinstanzen gegeben, die Operationen auf Kopien einer zu der vom Verfahren von

Thomas strukturäquivalenten Datenbasis überwachen. Sind zusätzlich die Modifikationsanforderungen konfliktfrei, so ergibt sich für jedes der Verfahren folgende Anzahl von Nachrichten, die zur Durchführung einer Anforderung benötigt werden, falls Quittungen des Nachrichtentransportsystems sowie Interaktionen Kontrollinstanz-Benutzer vernachlässigt werden:

- Verfahren von Johnson: n,
- Verfahren von Thomas: n+n/2,
- Verfahren von Holler: 2n(n-1).

Für große n wächst die Belastung des Gesamtsystems bei Verfahren mit exklusiver Sperrung sehr stark, zumal wenn Konflikte zwischen Anforderungen zusätzliche Verdrängungsnachrichten implizieren. Die Inkaufnahme einer solchen Belastung ist nur gerechtfertigt, wenn diese Verfahren in ihrer Leistung die Sortierverfahren wesentlich überragen.

Schon bezüglich der Komplexität der Datenbasisstruktur und der zugelassenen Operationen existieren für die Verfahren von Johnson und Thomas beträchtliche Einschränkungen. Das Verfahren von Johnson erlaubt zwar gegenüber dem Verfahren von Thomas eine dem Relationenmodell ähnliche Strukturierung der Datenbasis, muß dann aber zusätzliche n·(n-1) Synchronisationsnachrichten im Falle einer Lösch-Operation berücksichtigen. Das Verfahren von Thomas gestattet andererseits "funktionale" Änderungen, d.h. Änderungen der Art "X:=X+Y", die im Verfahren von Johnson nicht erlaubt sind. Für das Verfahren von Holler sind dagegen beliebig komplexe Datenbasisstrukturen und Operationen bzw. Operationsfolgen zugelassen.

Das Abstimmungsprinzip des Verfahrens von Thomas kann zu erheblichen Verzögerungen in der Bearbeitung von Anforderungen führen, da die Abstimmungen sequentiell und nicht parallel durch die Menge der Verwaltungsinstanzen erfolgt. Die Aktualität der Kopien kann darunter leiden; dies grenzt den Einsatzbereich dieses Verfahrens z.B. in Realzeitsystemen enger ein.

Mit der wichtigste Aspekt beim Leistungsvergleich ist die Tatsache, daß die Verfahren unterschiedliche Integritätsgrade gewährleisten.

Das Verfahren von Johnson kann lediglich sichern, daß nach Beendigung jeglicher Änderungstätigkeit die Kopien gegen den Zustand der Identität konvergieren. Dies verhindert jedoch nicht solche Zwischenzustände der Datenbasis, die vorgegebenen Integritätsanforderungen hinsichtlich der Beziehungen zwischen Werten unterschiedlicher Datenbasisobjekte widersprechen. Das Verfahren von Thomas umgeht diesen Nachteil, durchläuft jedoch nicht notwendig eine "vollständige" Zustandsänderungsfolge, da nicht alle Operationen zum Zuge kommen müssen.

Für Anwendungen in Realzeitsystemen reicht das nicht aus, da zusätzliche Bedingungen hinsichtlich der operationalen Integrität des Gesamtsystems zu berücksichtigen sind. Werden in einem verteilten Prozeßlenkungssystem z.B. dispositive Vorgabewerte redundant gehalten, da die Führung des Prozesses nach diesen Vorgabewerten von einem Rechner und die Optimierung des Prozesses auf der Basis dieser Vorgabewerte auf einem anderen Rechner bearbeitet werden, so würde - da Verzögerungen in der Nachrichtenübermittlung unvermeidbar sind - bei Verwendung des Verfahrens von Thomas der Optimierungsalgorithmus ggf. mit neuen Vorgabewerten weiterarbeiten, während der Prozeß selbst noch mit alten Werten geführt wird. Als Integritätsbedingung muß die Koordination der Vorgabewerteänderungen gefordert werden. Solchen Integritätsbedingungen genügen die Verfahren mit exklusiver Sperrung.

3.2. Das Basisprotokoll

3.2.1. Beschreibung des Protokolls

Die Überlegenheit des Verfahrens mit exklusiver Sperrung legt es nahe, dieses in entsemantisierter Form als Basisprotokoll für die Behandlung allgemeinerer Koordinationsaufgaben durch kooperierende Kontrollinstanzen einzusetzen. Die Lösung besteht in der Anwendung des Koordinationszyklus des Verfahrens von Holler für die Koordinierung von Aktivitäten, die im kritischen Abschnitt von den Kontrollinstanzen gleichzeitig auszuführen sind. Diese Aktivitäten sind als Koordinationsanforderungen aufzufassen, die die Initialisierung von Koordinationszyklen durch die Kontrollinstanzen auslösen und Abstimmungsvorgängen zu unterwerfen sind. Die konkreten Aktivitäten

ergeben sich z.B. aus den Anforderungen seitens der Sperrmechanismen; sie werden in den folgenden Kapiteln ausführlich erläutert.

Wegen der Wichtigkeit des Basisprotokolls als Grundlage der in dieser Arbeit entwickelten Verfahren zur Sicherung der operationalen Integrität in verteilten Datenbasen bei dezentraler Kontrollstruktur soll dieses im folgenden detailliert beschrieben werden. Ausgangspunkt ist die Modellierung der Kontrollinstanz als Warteschlangensystem mit einer Bedienungsstation und zwei Warteschlangen - siehe Bild 3.3.

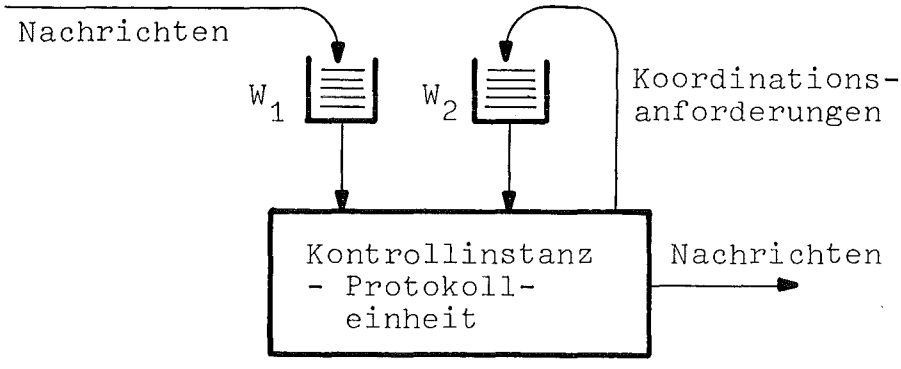


Bild 3.3: Modellierung einer Kontrollinstanz als Warteschlangensystem mit den Warteschlangen W_1 und W_2

Warteschlange W_1 nimmt die eingehenden Nachrichten auf. Wir unterscheiden interne Nachrichten, die Kontrollinstanzen untereinander austauschen, und externe Nachrichten, die das Eintreffen von Koordinationsanforderungen charakterisieren oder sich auf die Kommunikation der Kontrollinstanz mit dem lokalen Rechner beziehen. Warteschlange W_2 enthält die zu bearbeitenden Koordinationsanforderungen.

Der Kern der Bedienungsstation ist die Protokolleinheit. Sie ist für die Abwicklung des Koordinationszyklus und den damit verbundenen Nachrichtenaustausch zwischen den Kontrollinstanzen verantwortlich. Die Protokolleinheit läßt sich (in Anlehnung an /L2/) als sequentieller Automat durch das 7-Tupel

$$(S, I_{in}, I_{ex}, O_{in}, O_{ex}, M, N)$$

beschreiben, worin bedeuten:

S = Zustandsmenge

I_{in} = Alphabet der internen Eingabenachrichten

I_{ex} = Alphabet der externen Eingabenachrichten

O_{in} = Alphabet der internen Ausgabenachrichten

O_{ex} = Alphabet der externen Ausgabenachrichten

$M: S \times (I_{in} \cup I_{ex}) \rightarrow S$ die Zustandsübergangsfunktion

$N: S \times (I_{in} \cup I_{ex}) \rightarrow O_{in} \times O_{ex}$ die Ausgabefunktion

Für das konkrete Basisprotokoll ist die Protokolleinheit einer Kontrollinstanz bei einem System von n Kontrollinstanzen gegeben durch

$S = \{1,2,3,4\}$

$I_{in} = O_{in} = \{A_1,\ldots,A_n,E\}$

$I_{ex} = \{a,e\}$

$O_{ex} = \{d\}$

und den Graphen von Bild 3.4. Die Knoten entsprechen den Elementen von S, die Pfeile zusammen mit den beigeordneten Beschriftungen der Form

$$x_1,x_2/y_1,y_2$$

mit $x_1 \varepsilon I_{ex}$, $x_2 \varepsilon I_{in}$, $y_1 \varepsilon O_{ex}$, $y_2 \varepsilon O_{in}$ definieren die Funktionen M und N.

Mit den internen Nachrichten A_i, i=1(1)n, zeigen sich die Kontrollinstanzen gegenseitig die Bereitschaft an, die von der Kontrollinstanz i, i=1(1)n, akzeptierte Aktivität (Koordinationsanforderung) auszuführen, und mit der Nachricht E, daß sie die Aktivität beendigt haben. Die externe Nachricht a bedeutet die Ankunft einer Transaktion in der Kontrollinstanz. Mit dem Kommando d wird die Ausführung der Aktivität gestartet und mit e deren Beendigung angezeigt (d und e sind nur für ihre Protokolleinheit bestimmte private Anzeigen einer Kontrollinstanz).

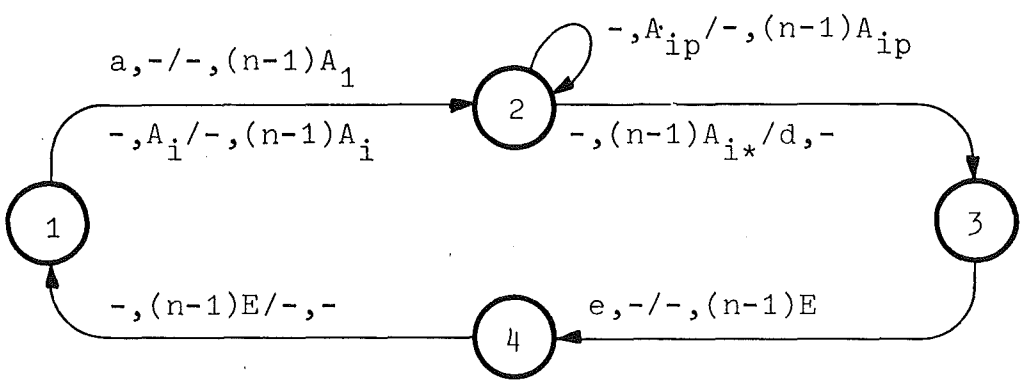


Bild 3.4: Basisprotokoll für Kontrollinstanz 1

Die Arbeitsweise der Kontrollinstanzen ist wie folgt (siehe hierzu auch Bild 3.5):

Zustand 1:
Ein Koordinationszyklus kann nur im Zustand 1 beginnen. Er kann durch Eigeninitialisierung (externe Nachricht a) oder Fremdinitialisierung (interne Nachricht vom Typ A) ausgelöst werden. Die Kontrollinstanz sendet entsprechende Bereitschaftserklärungen (interne Nachricht vom Typ A) an die restlichen n-1 Kontrollinstanzen und geht in den Zustand 2 über.

Eine Verdrängung von in Abstimmung befindlichen Koordinationsanforderungen ist im Zustand 1 möglich; siehe hierzu die Ausführungen unter Zustand 2.

Zustand 2:
In diesem Zustand wartet die Kontrollinstanz auf das Eintreffen von Bereitschaftserklärungen der restlichen n-1 Kontrollinstanzen. Nach Empfang von n-1 (akkumulierten) Bestätigungen für eine Anforderung A_{i*}, für die sie ihrerseits Bereitschaftserklärungen an die n-1 restlichen Kontrollinstanzen gesendet hat, wechselt sie sofort in den Zustand 3 über.

Eine Verdrängung von in Abstimmung befindlichen Koordinationsanforderungen durch eine Anforderung höherer Priorität A_{ip} ist einer Kontrollinstanz nur im Zustand 1 und 2 erlaubt. Beruht die Forderung nach Verdrängung auf Fremdinitialisierung, so kann eine Kontrollinstanz ihr nur zustimmen, falls sie noch keine n-1 Bestätigungen für eine Koordinationsanforderung empfangen hat. Sie selbst kann eine Verdrängung nur initialisieren, falls sie noch keine Bereitschaftserklärungen für eine andere Koor-

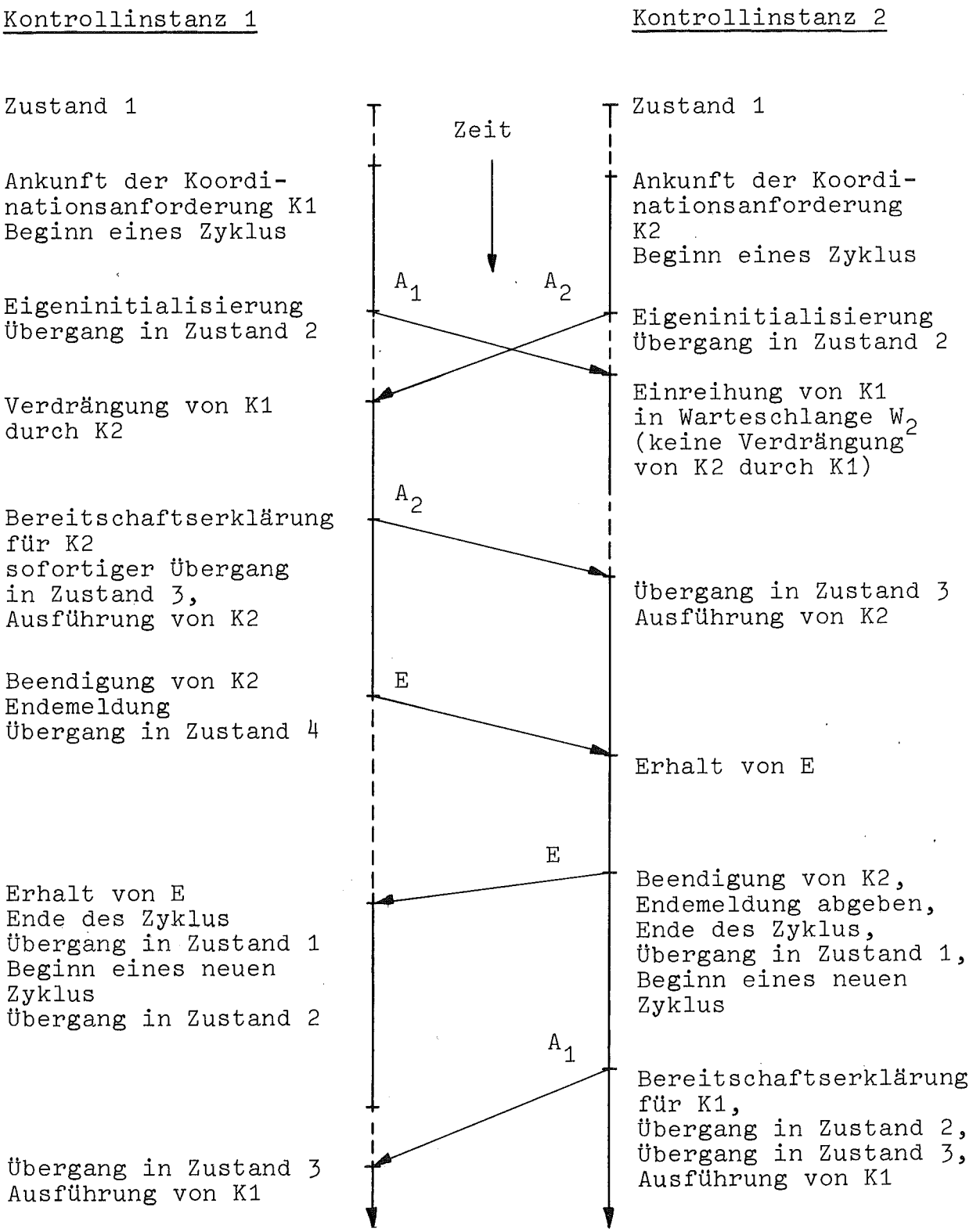


Bild 3.5: Beispiel für die Koordinierung von Aktivitäten durch das Basisprotokoll in einem System von zwei Kontrollinstanzen (----- Wartezeiten, ——— Aktivzeiten der Kontrollinstanzen)

dinationsanforderung an die restlichen Kontrollinstanzen abgesendet hat und selbst nicht Initiator einer anderen Koordinationsanforderung war.

Abgegebene Bereitschaftserklärungen für verdrängte Koordinationsanforderungen gehen nicht verloren, sondern werden diesen Koordinationsanforderungen für Abstimmungen in den folgenden Zyklen gutgeschrieben und ausgewertet.

Zustand 3:
In diesem Zustand wird die (allgemein akzeptierte) Aktivität ausgeführt; die Verdrängung einer in Ausführung befindlichen Aktivität (einer Koordinationsanforderung) ist nicht erlaubt.

Zustand 4:
Nach der Ausführung der Aktivität und der Benachrichtung der restlichen Kontrollinstanzen (Nachricht des Typs E) kehrt eine Kontrollinstanz erst in den Zustand 1 zurück, falls sie die entsprechenden Bereitschaftserklärungen von allen anderen n-1 Kontrollinstanzen vorliegen hat.

Nach Rückkehr in den Zustand 1 ist die Kontrollinstanz für einen neuen Zyklus bereit. Stehen weitere Koordinationsanforderungen zur Bearbeitung an, wird der neue Zyklus unter Beachtung der Prioritäts- und Verdrängungs-Regelung gestartet. Für eine verdrängte Koordinationsanforderung, für die sie in einem der früheren Zyklen eine Bereitschaftserklärung abgegeben hat, ist dies nicht zu wiederholen; der Übergang in den Zustand 2 kann dann unmittelbar ohne Aussendung von Bereitschaftserklärungen erfolgen.

Eine Vereinfachung des Basisprotokolls, die im weiteren ebenfalls benötigt wird, sieht bei Verdrängung die sofortige Elimination der für die verdrängten Koordinationsanforderungen abgegebenen Bereitschaftserklärungen vor, d.h. die Gültigkeit von Bereitschaftserklärungen ist auf den Zyklus beschränkt, innerhalb dessen sie abgegeben wurden.

Die Korrektheit des Basisprotokolls wurde in /H6/ für den Fall $n=2$ ausführlich nachgewiesen; für den Fall $n>2$ soll dies im folgenden geschehen.

3.2.2. Nachweis der Korrektheit des Basisprotokolls

Zum Nachweis der Korrektheit des Basisprotokolls ist zu zeigen:

1. Die Verklemmungsfreiheit der Kommunikation der Kontrollinstanzen
   - Bei Initialisierung eines Koordinationszyklus durch eine Kontrollinstanz hat nach endlicher Zeit ein Übergang aller Kontrollinstanzen in den Zustand 3, in dem die zu koordinierende Aktivität durchgeführt wird, zu erfolgen.
   - Initialisierungen von Koordinationszyklen durch verschiedene Kontrollinstanzen dürfen nicht zu einem Stillstand der Kooperation der Kontrollinstanzen führen, falls noch Koordinationsanforderungen zur Bearbeitung anstehen.
2. Die Konsistenz der Aktivitäten der Kontrollinstanz im Zustand 3: Nach Übergang der Kontrollinstanzen in den Zustand 3 während eines Zyklus dürfen die von den Kontrollinstanzen ausgeführten Aktivitäten nicht von unterschiedlichen Koordinationsanforderungen stammen.

Die Ausführung der Aktionen des Basisprotokolls durch die Kontrollinstanzen resultiert in einem System nebenläufiger, interagierender (sequentieller) Prozesse mit zyklischer Struktur. Als Darstellungsmittel für solche Systeme eignen sich Petri-Netze /L3/. Die Darstellung erfordert eine detaillierte Auflösung des Systemgeschehens in elementare Ereignisse und deren Wechselbeziehungen; sie kann daher als Grundlage für eine Implementierung dienen; für unsere Zwecke ist sie unnötig aufwendig.

Wir folgen einer Idee von Ellis /E1/ und benutzen als Darstellungsmittel Lindenmayer-Systeme (kurz:L-Systeme) /R1/, die sich ebenfalls zur Beschreibung von Systemen paralleler Prozesse eignen. L-Systeme entsprechen Phrasenstrukturgrammatiken mit folgenden Änderungen /R1/:

- es wird nicht zwischen terminalen und nicht-terminalen Symbolen unterschieden,
- in einem Ableitungsschritt werden alle auftretenden Symbole einer Zeichenreihe gleichzeitig ersetzt (parallel rewriting).

Wir benutzen in Anlehnung an /E1/ eine vereinfachte Definition. Seien k, l natürliche Zahlen; unter einem L-System mit Tabellen und <k,l> Interaktionen (kurz <k,l> TIL-System) verstehen wir ein 4-Tupel

$$G = (\Sigma, P, g, w),$$

worin bedeuten:

Σ = endliches, nichtleeres Alphabet von G,
g = Markierungssymbol (nicht in Σ),
w = ein Wort über Σ, das Axiom von G,
P = endliche, nichtleere Menge von Tabellen.

Jede Tabelle $p \in P$ ist eine endliche nichtleere Relation, die erfüllt:

$$p \subseteq \bigcup_{\substack{i,j,m,n \geq 0 \\ i+j=k \\ m+n=l}} \{g^i\}\ \Sigma^j \times \Sigma \times \Sigma^m\ \{g^n\} \times \Sigma^*$$

mit der Vollständigkeitsbedingung:

für jedes $\langle \alpha, a, \beta \rangle$ in $\bigcup_{\substack{i,j,m,n \geq 0 \\ i+j=k \\ m+n=l}} \{g^i\}\ \Sigma^j \times \Sigma \times \Sigma^m\ \{g^n\}$

gibt es ein γ in Σ^*, so daß $\langle \alpha, a, \beta, \gamma \rangle$ ein Element von p ist. (Es bedeuten: Σ^0 das Leerwort; Σ^j die j-stellige Relation über Σ für $j>0$; $\Sigma^* = \bigcup_{j=0}^{\infty} \Sigma^j$ die Menge aller Worte über Σ).

Jedes Element von p heißt eine Produktion und wird normalerweise geschrieben in der Form $\langle \alpha, a, \beta \rangle \rightarrow \gamma$. Sei mit "-" das leere Wort bezeichnet; eine kontextfreie Produktion $\langle -, a, - \rangle \rightarrow \gamma$ wird kurz $a \rightarrow \gamma$ geschrieben. Beim Spezifizieren von Produktionen in einer Tabelle sind diejenigen nicht zu berücksichtigen, die ausgehend vom Axiom w in keinem Ersetzungsprozeß benutzt werden können. Markierungssymbole in Produktionen können vernachlässigt werden, falls die Eindeutigkeit von Ersetzungsprozessen nicht verletzt wird.

Sei $x = a_1 \ldots a_n \in \Sigma^*$, mit $a_1, \ldots, a_n \in \Sigma$, und $y \in \Sigma^*$; y heißt direkt ableitbar von x in G (bezeichnet mit $x \underset{G}{\Rightarrow} y$), falls

$y = \gamma_1 \ldots \gamma_n$ für einige $\gamma_1,\ldots,\gamma_n \in \Sigma^*$, und es eine Tabelle p gibt, in der zu jedem a_i, i = 1(1)n, eine Produktion existiert der Form

$$\langle \alpha_i, a_i, \beta_i \rangle \rightarrow \gamma_i$$

wobei α_i das Präfix $g^j a_1 \ldots a_{i-1}$ der Länge k und β_i das Suffix $a_{i+1} \ldots a_n g^m$ der Länge l ist.

Die transitive und reflexive Hülle der Relation $x \underset{G}{\Rightarrow} y$ wird mit $x \overset{*}{\underset{G}{\Rightarrow}} y$ bezeichnet, und in diesem Fall heißt y ableitbar aus x in G.

Die Sprache, die durch G erzeugt wird, ist definiert durch $L(G) = \{x \in \Sigma^* : w \overset{*}{\underset{G}{\Rightarrow}} x\}$, d.h. der Menge aller aus dem Axiom w in G ableitbaren Worte über dem Alphabet Σ.

Betrachten wir nun einen Zyklus, so können wir den Zustand der Protokolleinheit i, i=1(1)n, bezüglich des Protokolls eindeutig beschreiben durch das geordnete Paar (s_i, t_i) mit

$s_i \in \{a_i, b_i, c_i, d_i\}$ als Element der Zustandsmenge S
($a_i \triangleq$ Zustand 1, $b_i \triangleq$ Zustand 2, $c_i \triangleq$ Zustand 3, $d_i \triangleq$ Zustand 4) und

$t_i \in \{A_1,\ldots,A_n, E, O\}$ (O = Leernachricht) als dem zuletzt an die anderen Kontrollinstanzen ausgesandten Nachrichtentyp.

Mögliche Zustände sind für eine Kontrollinstanz i, i=1(1)n, nur:

(a_i, O), $\{(b_i, A_j) : j = 1(1)n\}$, $\{(c_i, A_j) : j = 1(1)n\}$ und (d_i, E).

Die Nachrichten {d,e} werden nicht berücksichtigt, da sie keine Interaktionen zwischen Kontrollinstanzen widerspiegeln. Der Gesamtzustand eines Systems von n Kontrollinstanzen wird zu einem Zeitpunkt des Zyklus durch das geordnete n-Tupel

$$((s_1,t_1),\ldots,(s_n,t_n))$$

der möglichen Zustände (s_i,t_i) der Kontrollinstanzen i, i=1(1)n, beschrieben. Die Zustandsübergänge einer Kontrollinstanz können unabhängig oder abhängig von denen anderer Kontrollinstanzen stattfinden, im letzteren Falle aufgrund von protokollgesteuerten Interaktionen der Kontrollinstanzen.

Die Prioritäten der Koordinationsanforderungen seien durch die Identifikation der Kontrollinstanz gegeben, wobei für i=1(1)n-1 gelten soll:

Priorität (Kontrollinstanz i) < Priorität (Kontrollinstanz (i+1))

Dies stellt keine Einschränkung des allgemeinen Falls dar, in dem die Prioritätsregelung unabhängig von der Kontrollinstanzidentifikation erfolgt, da ein "idealer Beobachter" die Kontrollinstanzen ggf. umnumerieren kann, um die obige Prioritätsordnung zu erhalten.

Wir betrachten zunächst das vereinfachte Basisprotokoll, bei dem Bereitschaftserklärungen, die für verdrängte Koordinationsanforderungen an andere Kontrollinstanzen übermittelt wurden, vom System vergessen werden, d.h. also nicht in späteren Zyklen verwendet werden dürfen. Die notwendige Erweiterung zur formalen Beschreibung des Basisprotokolls, in dem diese Einschränkung nicht gilt, wird im Anschluß daran diskutiert.

Ein System von n Kontrollinstanzen kann unter diesen Voraussetzungen durch folgendes <n-1, n-1> TIL-System G = (Σ,P,g,w) beschrieben werden mit:

$\Sigma = \{\{(a_i,0) : i = 1(1)n\}, \{(b_i,A_j) : i, j = 1(1)n\},$
$\{(c_i,A_j) : i, j = 1(1)n\}, \{(d_i,E) : i = 1(1)n\}\}$

d.h. Σ besitzt als Elemente die möglichen Zustände der Protokolleinheiten; falls keine Verwechslungen zu befürchten sind, wird eine vereinfachte Schreibweise bevorzugt: statt (b_i,A_j) nur b_iA_j.

$w = (a_10)(a_20) \ldots (a_n0)$

d.h. alle Kontrollinstanzen befinden sich im Zustand 1 und warten auf den Eingang von Koordinationsanforderungen.

g = # ; die Rolle von g ist in diesem Fall unwesentlich, da die Ersetzung von Symbolen aus Σ aufgrund ihrer Semantik positionsgebunden vorzunehmen ist.

P besteht aus genau einer Tabelle mit folgenden Produktionsklassen:

1. $i = 1(1)n$: $a_i O \rightarrow a_i O$

2. Eigeninitialisierung
   $i = 1(1)n$: $a_i O \rightarrow b_i A_i$

3. Fremdinitialisierung
   $i = 1(1)n$: $\langle u_0\ u_1 \ldots u_{i-1}, u_i, u_{i+1} \ldots u_n\ u_{n+1}\rangle \rightarrow b_i A_j$
   wobei a) $u_0 = u_{n+1}$ = "-" (Leerwort)
   b) $u_i = a_i O$
   c) sei $M_i = \{m: m = 1(1)n, m \neq i\}$; es muß eine nicht leere Menge $R \subseteq M_i$ existieren mit $u_r = b_r A_r$, $r \varepsilon R$; für die u_l mit $l \varepsilon M_i - R$ muß gelten:
   $u_l = a_l O$ oder $u_l = b_l A_k$ mit $k \leq \max\{r \varepsilon R\}$
   d) in $b_i A_j$ (rechte Seite der Produktion) kann j beliebig aus der nicht leeren Menge
   $\{k: u_k = b_k A_k, k \leq \max\{r \varepsilon R\}\}$ gewählt werden.

4. Verdrängung
   $i = 1(1)n$: $\langle u_0 u_1 \ldots u_{i-1}, u_i, u_{i+1} \ldots u_n\ u_{n+1}\rangle \rightarrow b_i A_j$
   wobei a) $u_0 = u_{n+1}$ wie 3a)
   b) $u_i = b_i A_l$, $l \varepsilon \{1,\ldots,n\}$, und es muß mindestens ein $k \varepsilon \{1,\ldots,n\}$ existieren, für das gilt: $u_k = b_k A_k$ und $l \leq k$.
   c) wie 3c)
   d) sei $u_i = b_i A_l$, $l \varepsilon \{1,\ldots,n\}$;
   in $b_i A_j$ kann j beliebig aus der nichtleeren Menge $\{k: u_k = b_k A_k, k = 1(1)n, k > \max(l,i)\}$ gemäß vorausgesetzter Prioritätsfestlegung für Koordinationsanforderungen gewählt werden.

5. $i, j = 1(1)n$: $b_i A_j \rightarrow b_i A_j$

6. Übergang in den Zustand 3:
   $i, j = 1(1)n$: $\langle u_0 u_1 \ldots u_{i-1}, u_i, u_{i+1} \ldots u_n\ u_{n+1}\rangle \rightarrow c_i A_j$
   mit: a) wie 3a)
   b) $u_k = b_k A_j$ für $k = 1(1)n$

7. $i,j = 1(1)n : c_iA_j \rightarrow c_iA_j$

8. $i,j = 1(1)n : c_iA_j \rightarrow d_iE$

9. $i = 1(1)n : d_iE \rightarrow d_iE$

10. $i = 1(1)n : \langle u_0u_1\ldots u_{i-1}, u_i, u_{i+1}\ldots u_n\ u_{n+1}\rangle \rightarrow a_iO$

    mit a) wie 3a)

    b) $u_k = d_kE$ für $k = 1(1)n$

Für die Anwendung der Produktionen der Klassen 6 und 10 ist folgende Zusatzvorschrift zu berücksichtigen: Sei $u_1\ldots u_n \in \Sigma^*$; wird für ein u_i, $i \in \{1,\ldots,n\}$, eine Produktion aus einer dieser Klassen angewendet, so sind für alle u_k, $k=1(1)n$, Produktionen aus der entsprechenden Klasse anzuwenden. Diese (produktionensparende) Vorschrift ist sinnvoll, da die entsprechenden Übergänge in den Zustand 3 bzw 1 erst vollführt werden können, wenn alle Kontrollinstanzen hierzu ihre Zustimmung erteilt und die entsprechende Meldung ausgesendet haben.

Für den Fall n = 2 ist in Tabelle 3.1 die Produktionenmenge zusammengestellt; Bild 3.6 zeigt die Gesamtheit der möglichen Ableitungen.

Für den Nachweis der Korrektheit muß geprüft werden:

1. Ist die Kommunikation blockierungsfrei?
   Dies würde bedeuten, daß die kontextfreien Produktionen der Klassen 1, 5, 7 und 9 unendlich oft sukzessiv für eine oder mehrere Kontrollinstanzen zur Anwendung kommen. Wir setzen ein zuverlässiges Gesamtsystem, in dem Nachrichten innerhalb endlicher Zeit übertragen und von den Kontrollinstanzen bearbeitet werden, voraus und können daher die ununterbrochene Anwendung einer solchen Produktion auf eine endliche Zahl beschränken. Nach endlichmaliger Anwendung einer nicht den Zustand verändernden Produktion für eine Kontrollinstanz kommt eine den Zustand verändernde Produktion zur Anwendung.

2. Ist die Kooperation der Kontrollinstanzen verklemmungsfrei?
   Mit A(G) bezeichnen wir die Sprache für ein TIL-System G, die aus der Menge der Worte aus L(G) besteht, die aus sich und nur aus sich selbst (direkt) ableitbar sind (adult lan-

guage /R1/). Die Verklemmungsfreiheit der Kooperation der Kontrollinstanzen ist gesichert, falls A(G) die leere Menge ist.

Ein Beispiel /R1/: sei $G_1 = <\Sigma_1, P_1, w_1, g_1>$ mit $\Sigma_1 = \{a,b\}$, $P_1 = \{a \rightarrow$ "-" (leeres Wort), $a \rightarrow ab$, $b \rightarrow b\}$ und $w = a$; $A(G_1) = \{b^n : n \geq 0\}$.

Für das durch G dargestellte Basisprotokoll ist A(G) leer, da

- kontextfreie Produktionen nur endliche Male in ununterbrochener Reihenfolge angewendet werden dürfen,
- die Produktionen der Klassen 2, 3, 4, 6 und 10 für den Systemfortschritt sorgen und nicht in sich selbst überführen; in Klasse 4 wird dies durch die Prioritätsregelung gesichert.

3. Ist die Konsistenz der Aktivitäten der Kontrollinstanz in Zustand 3 gesichert?
   Die Konsistenz ist verletzt, falls L(G) Worte $u_1 \ldots u_n$ enthält, derart daß $u_i = c_i A_{j_i}$, $i = 1(1)n$, $j_i \in \{1,\ldots,n\}$, und es gibt mindestens zwei u_i, u_k, $k \neq i$, mit $u_i = c_i A_j$ und $u_k = c_k A_1$ mit $j \neq 1$.
   Die Entstehung solcher Worte wird aber durch die Produktionen der Klassen 4 und 6 verhindert.

Damit ist die Korrektheit des vereinfachten Basisprotokolls nachgewiesen.

Abschließend sei die Einschränkung aufgehoben, daß Bereitschaftserklärungen, die während eines Zyklus für verdrängte Koordinationsanforderungen abgegeben wurden, in nachfolgenden Zyklen nicht verwendet werden durften. Wir erweitern hierzu die bisherigen lokalen Zustände (s_i, t_i) einer Kontrollinstanz i um den Vektor $z_i = (z_{i1},\ldots,z_{in})$ zu (s_i, t_i, z_i), welche nun als Elemente von Σ gelten.

Für die Komponenten von z_i, $i = 1(1)n$, gilt:

$$z_{ij} = \begin{cases} 1 & \text{falls von Kontrollinstanz i für eine von j initialisierte Koordinationsanforderung Bereitschaftserklärungen abgesendet wurden und die Koordinationsanforderung noch nicht bearbeitet ist.} \\ 0 & \text{sonst} \end{cases}$$

Tabelle der Produktionen:

1.	$a_1O \rightarrow a_1O$	$a_2O \rightarrow a_2O$
2.	$a_1O \rightarrow b_1A_1$	$a_2O \rightarrow b_2A_2$
3.	$\langle -, a_1O, b_2A_2\rangle \rightarrow b_1A_2$ $\langle b_1A_1, a_2O, -\rangle \rightarrow b_2A_1$	
4.	$\langle -, b_1A_1, b_2A_2\rangle \rightarrow b_1A_2$	
5.	$b_1A_1 \rightarrow b_1A_1$ $b_1A_2 \rightarrow b_1A_2$	$b_2A_1 \rightarrow b_2A_1$ $b_2A_2 \rightarrow b_2A_2$
6.	$\langle -, b_1A_1, b_2A_1\rangle \rightarrow c_1A_1$ $\langle -, b_1A_2, b_2A_2\rangle \rightarrow c_1A_2$ $\langle b_1A_2, b_2A_2, -\rangle \rightarrow c_2A_2$ $\langle b_1A_1, b_2A_1, -\rangle \rightarrow c_2A_1$	
7.	$c_1A_1 \rightarrow c_1A_1$ $c_1A_2 \rightarrow c_1A_2$	$c_2A_1 \rightarrow c_2A_1$ $c_2A_2 \rightarrow c_2A_2$
8.	$c_1A_1 \rightarrow d_1E$ $c_1A_2 \rightarrow d_1E$	$c_2A_1 \rightarrow d_2E$ $c_2A_2 \rightarrow d_2E$
9.	$d_1E \rightarrow d_1E$	$d_2E \rightarrow d_2E$
10.	$\langle -, d_1E, d_2E\rangle \rightarrow a_1O$ $\langle d_1E, d_2E, -\rangle \rightarrow a_2O$	

Tabelle 3.1: Tabelle der Produktionen für ein System von zwei Kontrollinstanzen bei Anwendung des vereinfachten Basisprotokolls

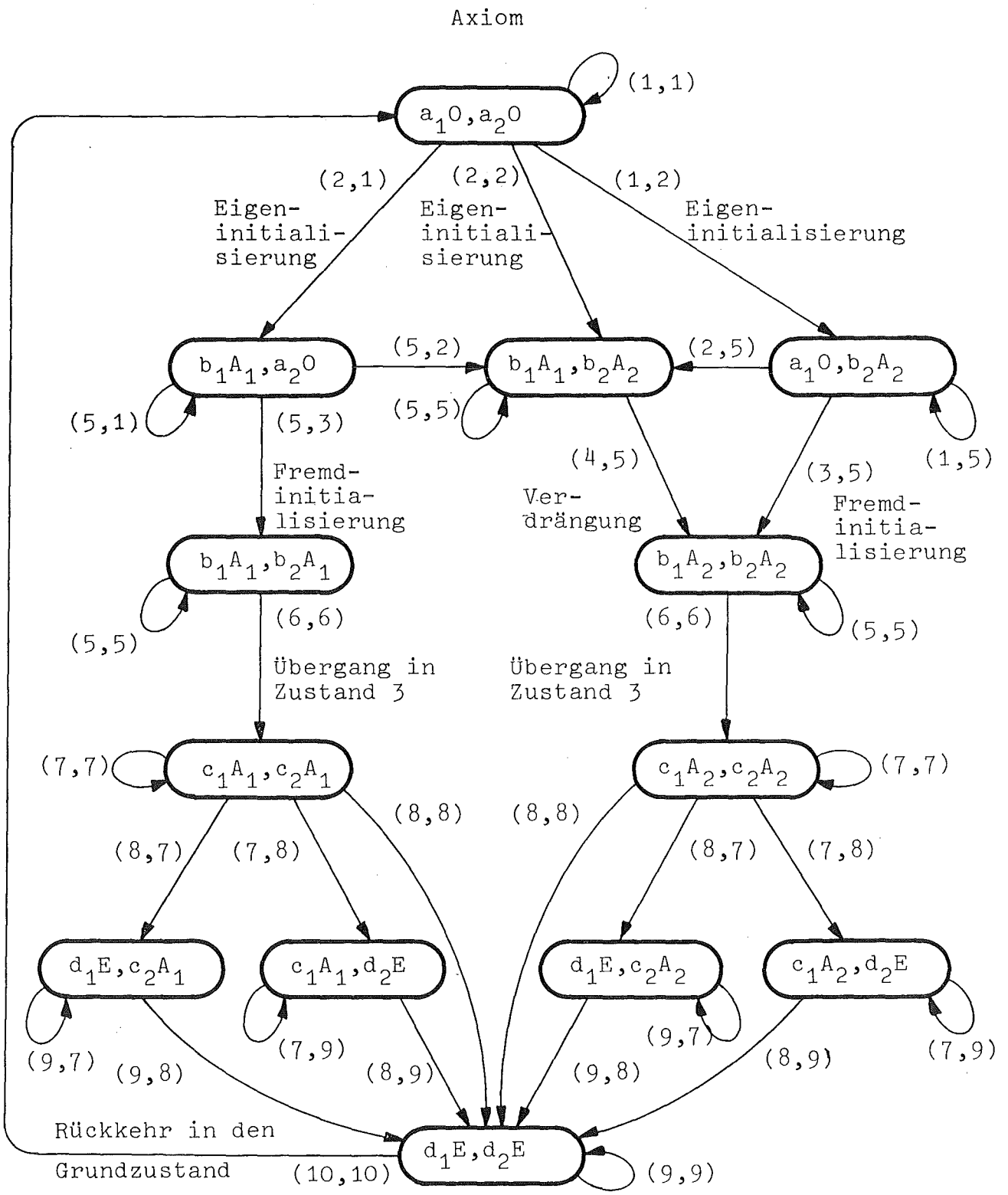


Bild 3.6: Zusammenfassung der durch die Produktionen von Tabelle 3.1 ermöglichten Ableitungen für ein System von zwei Kontrollinstanzen bei Anwendung des vereinfachten Basisprotokolls (die Ziffern in Klammern weisen auf die bei der Ableitung angewendeten Produktionsklassen hin).

Als neues Axiom von G dient

$$(a_1 0\bar{0},\ a_2 0\bar{0},\ldots,\ a_n 0\bar{0})$$

mit $\bar{0}$ als Bezeichnung für den Nullvektor.

Zur Einarbeitung von z_i, i = 1(1)n, in die Produktionen von P sind insbesondere die Produktionen der Klassen 2, 3, 4, 8 und 10 zu berücksichtigen, da nur diese z_i, i = 1(1)n, verändern.

In den Produktionen der Klassen 2, 3, 4 wird z_{ij} entsprechend dem in der rechten Seite der Produktion - $b_i A_j$ - auftretenden Index j von A_j auf 1 erhöht, in den Produktionen von Klasse 8 wird nach Ausführung von Koordinationsanforderung j z_{ij} auf 0 gesetzt.

Die Produktionen von Klasse 10 sind neu festzulegen:

i = 1(1)n:

1. $\langle u_0 u_1 \ldots u_{i-1},\ u_i,\ u_{i+1} \ldots u_n\ u_{n+1} \rangle \rightarrow a_i 0\bar{0}$

   mit a) wie 3a)

   b) $u_k = d_k E z_k$ für k = 1(1)n

   c) $z_i = \bar{0}$

2. $\langle u_0 u_1 \ldots u_{i-1},\ u_i,\ u_{i+1} \ldots u_n\ u_{n+1} \rangle \rightarrow b_i A_j z_i$

   mit a) wie 3a)

   b) $u_k = d_k E z_k$ für k = 1(1)n

   c) es gibt ein l ε {1,...,n} mit $z_{il} = 1$

   d) j = max {k: k ε {1,...,n} und $z_{ik} = 1$}

Diese Produktionen reflektieren die Forderung, daß nach Abgeben einer Bereitschatfserklärung für eine Koordinationsanforderung eine Kontrollinstanz solange das Recht auf Eigeninitialisierung verloren hat, bis diese Koordinationsanforderung ausgeführt worden ist.

Die Korrektheit des erweiterten Protokolls ist gesichert, da die erweiterten Produktionen der Klassen 1 - 9 nicht zu globalen Interaktionen führen, sondern das vereinfachte Basisprotokoll strikt einhalten und die Produktionen von Klasse 10 den

Übergang in genau solche Zustände implizieren, die im TIL-System für das vereinfachte Basisprotokoll als erlaubt gelten.

4. Sperrmechanismen bei dezentraler Kontrollstruktur und zuverlässigem Gesamtsystem

4.1. Struktur einer Kontrollinstanz

Das in Kapitel 3 vorgestellte Basisprotokoll kann unmittelbar verwendet werden, um die Bearbeitung von Transaktionen systemweit zu koordinieren, wenn als Koordinationsanforderung die vollständige Ausführung einer Transaktion interpretiert wird. Während der Ausführung einer Transaktion befinden sich alle Kontrollinstanzen im Zustand 3 und können diesen erst nach Transaktionsbeendigung verlassen. Diese systemweit serielle Abarbeitung von Transaktionen ist eine sehr ineffiziente Lösung, da eine Transaktion i.a. durch ihren Konsistenzbereich nicht die gesamte Datenbasis, sondern nur Teile davon beansprucht. Zur Erhöhung des Ausnutzungsgrades der Kapazität des verteilten DV-Systems ist zu fordern, daß Transaktionen, die sich nicht gegenseitig stören, parallel ausgeführt werden.

Bei der Konzeption, in der jede Kontrollinstanz den Zugriff auf genau eine Konstituente der Datenbasis überwacht, beinhaltet diese Forderung, daß eine Koordinierung von Teilmengen der Menge der Kontrollinstanzen möglich sein muß. Dieser Fall kann in einem allgemeineren Fall eingebettet werden, bei dem aus Effizienzgründen eine Kontrollinstanz für mehrere Konstituenten zuständig ist, z.B. wird man in einem Rechner eines Rechnernetzes genau eine Kontrollinstanz führen, die für alle lokal vorhandenen Konstituenten verantwortlich ist.

Bei zuverlässigem Gesamtsystem (alle Hardware- und Softwarekomponenten einschließlich Transaktionen arbeiten korrekt) und einer Betriebsführung, die eine parallele Ausführung von Transaktionen unterstützen soll, reduziert sich die Forderung nach der operationalen Integrität auf das Bereitstellen eines globalen Sperrmechanismus, der den Transaktionen das Arbeiten in ihren Konsistenzbereichen ohne gegenseitige Störungen sichert. Der globale Sperrmechanismus hat die von der Umwelt vorgegebenen Interdependenzen von Transaktionen zu berücksichtigen; um hiervon unabhängig zu sein, sei im folgenden vorausgesetzt, daß bei serieller Ausführung der Transaktionen in beliebiger Reihenfolge die operationale Integrität der Datenbasis gewährleistet ist.

Ein dezentral organisierter globaler Sperrmechanismus läßt sich nach dem Modell der dezentralen Kontrolle von Kap. 1 und den Ausführungen von Kap. 2 und 3 realisieren, in dem jede Kontrollinstanz - siehe Bild 4.1 - folgende Komponenten erhält:

- ein lokaler Sperrmechanismus zur Sicherung der operationalen Integrität der der Kontrollinstanz zugeordneten Konstituenten,
- eine Protokolleinheit zur Koordinierung der Aktivitäten der Kontrollinstanzen (insbesondere der Aktionen lokalen Sperrmechanismen) zur Transaktionsbearbeitung und zur Sicherung der operationalen Integrität der gesamten verteilten Datenbasis.

Zur globalen Koordinierung der Transaktionsbearbeitung benötigen die Kontrollinstanzen Information über den aktuellen Zustand der Datenbasis, der Transaktionsbearbeitung und des Protokolls.

Zur optimalen Transaktionsbearbeitung wären in der Kontrollinstanz zusätzlich ein Auftragsvergabealgorithmus und die zugehörige Zustandsinformation über den Belastungszustand des verteilten DV-Systems zu führen (siehe hierzu /H3/).

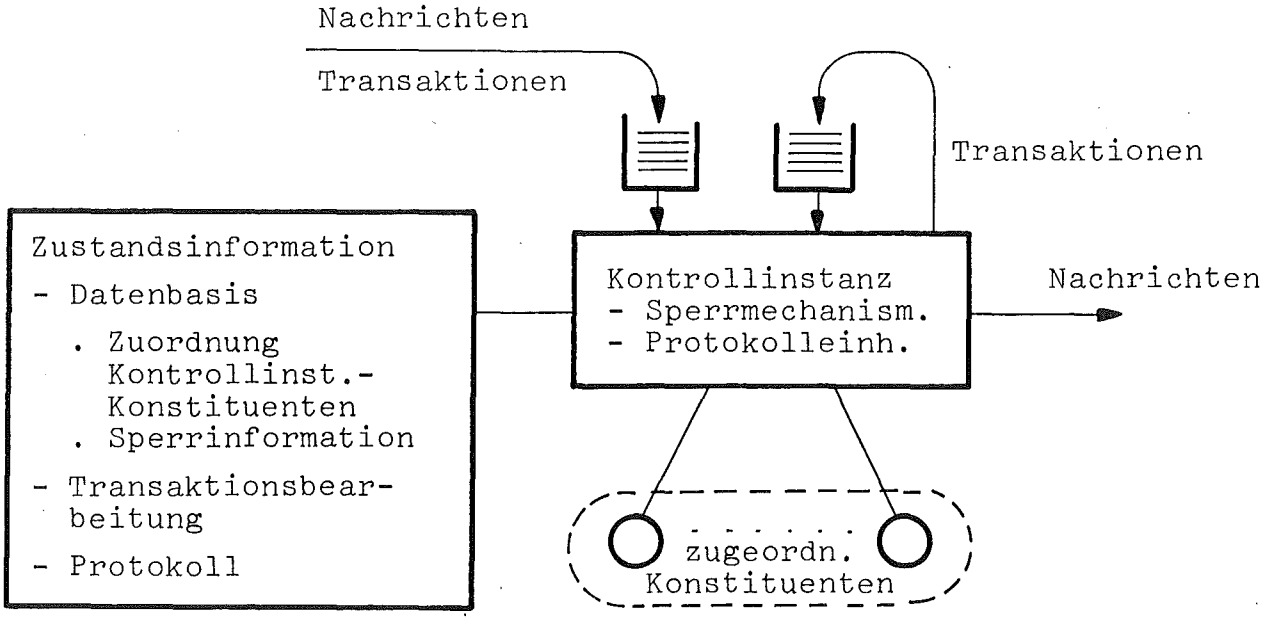


Bild 4.1: Prinzipielle Struktur einer Kontrollinstanz

Der globale aktuelle Zustand der verteilten Datenbasis setzt sich zusammen aus

- der Zuordnung von Konstituenten und Kontrollinstanzen,
- der Sperrinformation (aktuelle Sperrungsstufe der Sperreinheiten, aktuelle Zuordnung der Sperreinheiten zu den Transaktionen).

Der globale Zustand der Transaktionsbearbeitung umfaßt

- den aktuellen Bearbeitungszustand einer jeden Transaktion; dieser ist durch die Gesamtheit der Phasen (siehe 2.3) beschrieben, die die Transaktion bis zum aktuellen Zeitpunkt durchlaufen hat oder innehält, d.h. neben der aktuellen Phase ist auch die Geschichte einer Transaktion zu berücksichtigen,
- die Gesamtheit der Transaktionsspezifikationen bzgl. Betriebsmittelanforderungen,
- die Wartebeziehungen zwischen Transaktionen bzgl. Sperreinheiten.

Die von den Kontrollinstanzen zu bearbeitenden Koordinationsanforderungen resultieren im wesentlichen aus der Art der lokalen Sperrmechanismen und den Anforderungen seitens der Transaktionen zur Durchführung von Phasenübergängen, die mehrere Kontrollinstanzen gleichzeitig betreffen.

Unterschiedliche Verfahren für die Kooperation der Kontrollinstanzen ergeben sich aus der Kenntnis einer jeden Kontrollinstanz über den aktuellen globalen Zustand der verteilten Datenbasis und der Transaktionsbearbeitung.

Wir unterscheiden zwei Fälle:

1. Jede Kontrollinstanz besitzt die vollständige Kenntnis über diesen aktuellen globalen Zustand.
2. Jede Kontrollinstanz führt nur eine Teilsicht dieses aktuellen globalen Zustands.

In beiden Fällen sind zusätzliche Unterfälle zu berücksichtigen, die sich aus dem jeweils angewandten Sperrmechanismus ergeben.

Wir setzen eine feste Anzahl von Sperreinheiten voraus, d.h. Hinzufügen und Eliminieren von Sperreinheiten sind nicht erlaubt (siehe hierzu 4.5.).

4.2. Verfahren auf der Basis vollständiger Kenntnis des globalen Zustands

Die globale Zustandsinformation, aufgrund der die Kontrollinstanzen die Transaktionsbearbeitung leiten, liegt bei jeder Kontrollinstanz als Kopie vor. Eine verklemmungsfreie Kooperation der Kontrollinstanzen ist daher nur gewährleistet, wenn Veränderungen der Zustandsinformation nicht zu Konsistenzverletzungen dieses Kopien-Systems durch die Kontrollinstanzen führen. Dies erfordert, daß der lokale Sperrmechanismus in allen Kontrollinstanzen gleich ist und die Aktivitäten der Kontrollinstanzen in gegenseitiger Abstimmung erfolgen. Ein Protokoll, das letzteres leistet, kann durch geeignete Erweiterung des in 3.2 vorgeschlagenen Basisprotokolls erhalten werden.

4.2.1. Ein elementares Verfahren

Zur Vereinfachung setzen wir voraus, daß Transaktionen nach einer Ausführungsphase keine weiteren Sperrungen beantragen dürfen, d.h. Übergänge von Phase 4 nach Phase 2 sind nicht erlaubt und damit auch keine "sukzessiven Betriebsmittelanforderungen" (siehe hierzu 4.2.3.).

Die Protokolleinheit $(S, I_{in}, I_{ex}, O_{in}, O_{ex}, M, N)$ einer Kontrollinstanz i, i = 1(1)n, ist durch

$$S = \{1,2,3,3',4\}$$
$$I_{in} = \{A_1,\ldots,A_n,E\}$$
$$I_{ex} = \{a,e,v\}$$
$$O_{in} = I_{in}$$
$$O_{ex} = \{d\}$$

und die in Bild 4.2 dargestellten Funktionen M und N gegeben.

Mit den Nachrichten des Typs A_i, i = 1(1)n, zeigen sich die Kontrollinstanzen gegenseitig die Bereitschaft an, die von Kontrollinstanz i akzeptierte Aktivität durchzuführen. Die Aktivität ist in der Nachricht zu spezifizieren.

Unter den o.a. Einschränkungen müssen sich die Kontrollinstanzen bezüglich einer Transaktion für die Aktivitäten ihres lo-

kalen Sperrmechanismus

- Belegen der Betriebsmittel (assoziiert damit ist der Start der Ausführungsphase einer Transaktion durch die in ihre Ausführung involvierten Kontrollinstanzen),
- Freigeben von Betriebsmitteln (Transaktionsende),

koordinieren.

Eine Nachricht des Typs A umfaßt daher zumindest:

- Name der sendenden Kontrollinstanz
- Art der Aktivität (bei Belegung ist die Betriebsmittelliste beizufügen),
- Name der Transaktion,
- Name der initiierenden Kontrollinstanz.

Mit der Nachricht E informieren sich die Kontrollinstanzen darüber, daß sie die Aktivität im Zustand 3 bzw. 3' beendet haben; ausgelöst wird dies durch die (private) Anzeige e. Die externe Nachricht a bedeutet die Ankunft einer Transaktion. Mit dem Kommando d wird einer Transaktion der Start ihrer Ausführungsphase befohlen, die ihrerseits mit der Nachricht v der Kontrollinstanz meldet, daß alle gesperrten Objekte freigegeben werden können.

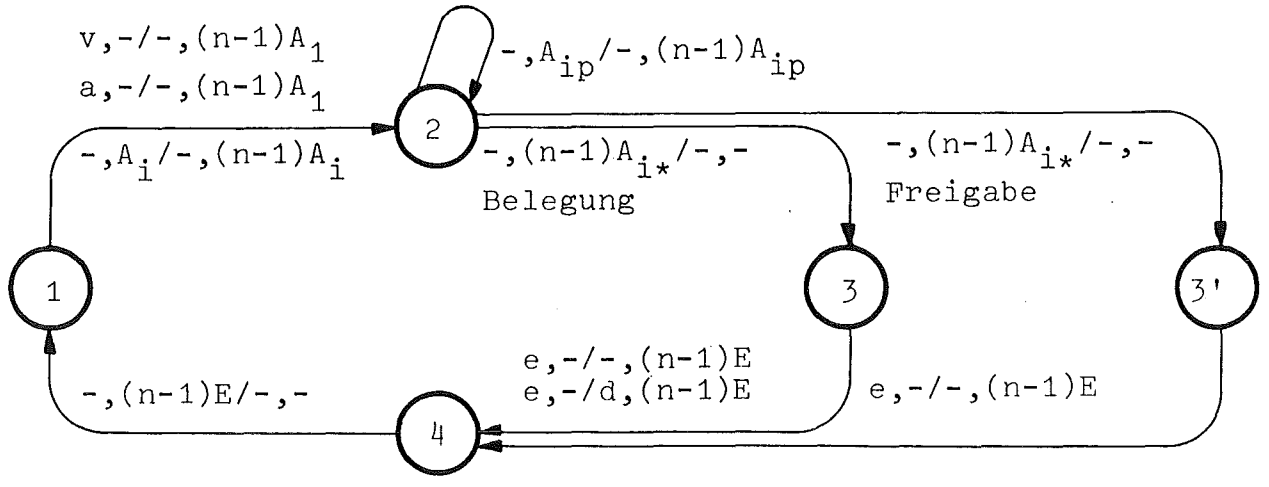


Bild 4.2: Protokoll für Kontrollinstanz 1

Der Arbeitsweise der Kontrollinstanzen wird das Basisprotokoll von 3.2 mit folgenden Zusätzen zugrundegelegt (vgl. 4.4.1.):

Zustand 1:
Die Eigeninitialisierung eines Zyklus für eine Aktivität "Belegung von Betriebsmitteln für eine Transaktion T" ist einer Kontrollinstanz nur erlaubt, falls alle von T benötigten Sperreinheiten gemäß vorliegender Sperrinformation kollisionsfrei (auf einmal) für T gesperrt werden können. Dies ist von der Kontrollinstanz zu prüfen. Diese Prüfung ist von einer Kontrollinstanz insbesondere auch für eine im vorangegangenen Zyklus verdrängte und im aktuellen Zyklus an erster Stelle stehende Aktivität dieser Art durchzuführen, falls das erweiterte Basisprotokoll von 3.2 zur Anwendung kommt, welches die Verwendung von abgegebenen Bereitschaftserklärungen für verdrängte Koordinationsanforderungen in darauffolgenden Zyklen erlaubt. Kann die Aktivität nicht durchgeführt werden, sind die abgegebenen Bereitschaftserklärungen als ungültig zu betrachten.

Zustand 2:
Die Verdrängungsregelung wird wie im Basisprotokoll gehandhabt. Zur Auflösung von Konfliktsituationen durch konkurrierende Koordinationsanforderungen hinsichtlich von Aktivitäten gleicher oder unterschiedlicher Art ist eine Prioritätsregelung einzuführen, z.B.

- zur Unterscheidung gleichartiger Aktivitäten: für i = 1(1)n-1, Priorität (Kontrollinstanz i) < Priorität (Kontrollinstanz i+1),
- zur Unterscheidung der Aktivitäten Belegung - Freigabe: Priorität (Belegung) < Priorität (Freigabe), um eine bessere Systemauslastung zu erzielen.

Der Übergang von Zustand 2 in den Zustand 3 bzw. 3' ist wie im Basisprotokoll geregelt und erfolgt in Abhängigkeit von der durchzuführenden Aktivität, die während ihrer Ausführung nicht verdrängt werden darf.

Zustand 3, Zustand 3':
Im Zustand 3 werden die Betriebsmittel für die jeweilige Transaktion gesperrt und die Zustandinformation entsprechend modifiziert. Danach kann die Transaktionsausführung gestartet werden, falls die der Kontrollinstanz zugehörigen Konstituenten

im Sperrbereich der Transaktion enthalten sind.

Transaktionen werden während ihrer Ausführungsphase nicht verdrängt.

Im Zustand 3' wird die mit einer Transaktionsbeendigung verbundenen Änderungen der Zustandsinformation durchgeführt.

Zustand 4:
Nach Durchführung der Aktivitäten in Zustand 3 bzw. 3' kehren die Kontrollinstanzen gemäß Basisprotokoll in den Zustand 1 zurück und sind für einen neuen Koordinationszyklus bereit.

Durch das Verfahren wird eine verklemmungsfreie Koordination der Transaktionsbearbeitung erzielt und die operationale Integrität der verteilten Datenbasis gesichert:

- Das Basisprotokoll einschließlich der Konfliktregelung sorgt für eine verklemmungsfreie Koordination der im Zustand 3 bzw. 3' durchzuführenden Aktivitäten und mit der Identität der Sperrmechanismen zusammen für die interne und externe Konsistenz der globalen Zustandsinformation.
- Verklemmungen von Transaktionen werden verhindert, in dem die von einer Transaktion benötigten Sperreinheiten dieser auf einmal zugewiesen werden, wenn sie in ihrer Gesamtheit kollisionsfrei zur Verfügung stehen (sukzessive Sperrung ist nicht erlaubt).

Um unendliches Warten von Transaktionen durch Verdrängungen zu vermeiden, ist in der Prioritätsregelung zusätzlich die Wartezeit von Transaktionen zu honorieren.

4.2.2. Verbesserung des elementaren Verfahrens

Zur Erhöhung des Parallelitätsgrades in der Transaktionsbearbeitung und zur Reduktion der Anzahl von Koordinationszyklen empfiehlt es sich, das o.a. Verfahren zu modifizieren, indem innerhalb der Abstimmungsphase (Zustände 1 und 2) Akkumulationen von Koordinationsanforderungen zugelassen werden. Bei Eigeninitialisierung darf eine Kontrollinstanz eine Menge von Koordinationsanforderungen der Art Belegung - Freigabe als "zusammengesetzte" Koordinationsanforderung präsentieren. In Bereit-

schaftserklärungen, die zur Beantwortung von Fremdinitialisierungen oder Verdrängungsanforderungen ausgesendet werden, ist die Gesamtheit der Koordinationsanforderungen zu spezifizieren, die ihr als in diesem Zyklus initialisiert bekannt geworden sind.

Bei Eintritt in den Zustand 3 - der Zustand 3' entfällt bei diesem Verfahren - ist allen Kontrollinstanzen die Gesamtheit der durchzuführenden Aktivitäten bekannt, die nun in ihrer durch die Prioritätsregelung vorgegebenen Reihenfolge ausgeführt werden, also Freigabeanforderungen vor Belegungsanforderungen. Belegungsanforderungen, die nicht erfüllt werden können, müssen in späteren Koordinationszyklen erneut beantragt werden.

Dieses Verfahren schließt die Wiederverwendung von Bereitschaftserklärungen in den weiteren Koordinationszyklen aus.

4.2.3. Sukzessive Betriebsmittelanforderungen

Die Verfahren sind zu erweitern, falls Transaktionen während einer Ausführungsphase (Phase 4) weitere Betriebsmittel zur Sperrung anfordern dürfen (Übergang in Phase 2). Prinzipiell sind zwei Vorgehensweisen gemäß den Strategien K1 (Verklemmungsverhinderung), K2 (Verklemmungsbeseitigung) von 2.1. denkbar, um dieses Problem zu behandeln.

Eine auf Verklemmungsverhinderung basierende Vorgehensweise setzt voraus, daß eine Transaktion ihre sukzessiv benötigten Betriebsmittelportionen vor Transaktionsbeginn bekanntgibt und das System ihr eine Anwartschaft auf die Gesamtanforderung bei Verfügbarkeit gutschreibt, die Betriebsmittel aber nur portionsweise zuteilt.

Eine auf der Strategie K2 basierende Vorgehensweise läßt Verklemmungen zu. Hierzu ist eine weitere Koordinationsanforderung zu berücksichtigen, nämlich der Antrag auf Durchführung des Algorithmus zur Entdeckung und Beseitigung von Verklemmungen. Die entsprechende Aktivität ist beim elementaren Verfahren in dem neu hinzufügenden Zustand 3'' bzw. im Zustand 3 beim verbesserten elementaren Verfahren auszuführen. Eine solche Anforderung ist in ihrer Priorität zweckmäßigerweise zwischen Freigabe und Belegung einzuordnen. Bei Freigabe von Betriebsmitteln ist an-

schließend zu prüfen, ob blockierte Transaktionen wieder entblokkiert werden können.

4.3. Verfahren auf der Basis eingeschränkter Kenntnis des globalen Zustands

Die bei einer Kontrollinstanz vorliegende Information über den globalen Zustand umfasse nur

- die existierenden Konstituenten und ihre Zuordnung zu den Kontrollinstanzen (diese Information benötigt die Kontrollinstanz zur Ermittlung der an einer Transaktionsbearbeitung zu beteiligenden Kontrollinstanzen),
- den aktuellen Zustand der ihr zugehörenden Sperreinheiten und deren Zuordnung zu in Bearbeitung befindlichen Transaktionen,
- den Bearbeitungszustand aller Transaktionen, in deren Bearbeitung sie involviert ist.

Die feste Kopplung der Aktionen aller Kontrollinstanzen in den Verfahren von 4.2. war durch die Forderung nach Konsistenzhaltung der bei den Kontrollinstanzen vorliegenden globalen Zustandsinformationen bedingt. Für die Sperrinformation entfällt nun diese Forderung, insbesondere müssen lokal bearbeitbare Transaktionen nicht mehr der globalen Koordinierung unterworfen werden (generell ist die gegenseitige Unabhängigkeit von Transaktionen bezüglich der Reihenfolge ihrer Bearbeitung vorausgesetzt). Dies läßt eine mehr entkoppelte Arbeitsweise der Kontrollinstanzen zu, bei der Zustandsübergänge und Sperrinformationsveränderungen nicht immer in globaler Abstimmung stattzufinden haben; sogar die Sperrmechanismen der einzelnen Kontrollinstanzen können differieren. Wir nennen daher diese Verfahren auch Verfahren mit loser Kopplung im Gegensatz zu denen mit fester Kopplung von 4.2.

Die Koordination der Phasen einer Transaktion kann sich auf die zu ihrer Bearbeitung erforderlichen Kontrollinstanzen beschränken. Für Transaktionen, deren Konsistenzbereich (und damit auch Sperrbereich) Konstituenten mehrerer Kontrollinstanzen umfassen, unterscheidet man zwischen

- dem lokalen, auf eine Kontrollinstanz bezogenen Bearbeitungszustand,
- dem globalen Bearbeitungszustand.

Gegenstand der Koordination für eine Transaktion ist die Überführung vom lokalen in den globalen Bearbeitungszustand.

Da die Konsistenzbereiche Konstituenten mehrerer Kontrollinstanzen einschließen können und eine Kontrollinstanz nicht den vollständigen globalen Zustand kennt, sind Situationen systemweiter gegenseitiger Blockierungen von Transaktionen möglich; diese erfordern ein gemeinsames, fest gekoppeltes Handeln von Kontrollinstanzen.

Hieraus kann folgende Grobstruktur für ein Protokoll in einem System von lose gekoppelten Kontrollinstanzen abgeleitet werden - siehe Bild 4.3 -:

- äußerer Zyklus:
  Diese Zustandsübergänge erfolgen in Abstimmung mit allen Kontrollinstanzen und entsprechen einer festen Kopplung der Kontrollinstanzen (problemorientierte Anwendung des Basisprotokolls),
- innerer Zyklus:
  Diese Zustandsübergänge erfolgen nicht in Abstimmung mit anderen Kontrollinstanzen.

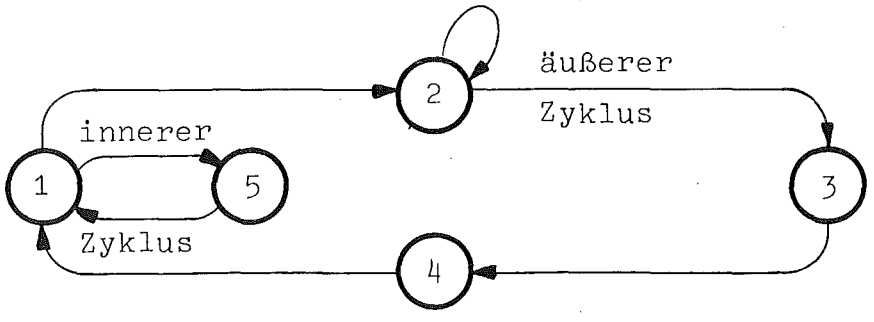


Bild 4.3: Grobstruktur eines Protokolls in einem System lose gekoppelter Kontrollinstanzen

Unterschiedliche Interpretationen der Aktionen im inneren und äußeren Zyklus der Protokolleinheit ergeben sich aus der gewählten Vorgehensweise zur globalen Regelung von Verklemmungssituationen (Strategienklassen K1, K2 in 2.1.):

- Verhindern von globalen Verklemmungen,
- Zulassen, Erkennen und Beseitigen von globalen Verklemmungen.

4.3.1. Verhindern globaler Verklemmungen

Wir setzen zunächst die Einschränkung voraus, daß eine Transaktion nach einer Ausführungsphase keine weiteren Betriebsmittel zur Sperrung anfordern (Verbot des Übergangs von Phase 4 nach Phase 2) darf.

In Anlehnung an die Diskussion der Strategien zur Verhinderung von Verklemmungen unter Berücksichtigung der speziellen Natur von Datenbasisobjekten als Betriebsmittel sowie dieser Einschränkung ergeben sich zwei Strategien, deren Einsatz zur globalen Transaktionskoordination sinnvoll erscheint:

- das globale Vorsortieren von Transaktionen,
- der Entzug aller für eine Transaktion reservierten Betriebsmittel, sobald sie blockiert wird; dies ist ein Sonderfall von 4.3.2. und wird dort bearbeitet.

Das Verfahren auf der Basis einer globalen Vorsortierung von Transaktionen beruht auf folgender Vorgehensweise:

1. Transaktionen, deren Konsistenzbereich Konstituenten mehrerer Kontrollinstanzen enthält, werden in eine systemweit gültige globale Rangordnung eingeordnet, die für alle Kontrollinstanzen zwingend eine Reihenfolge für die Abarbeitung der Transaktionen vorschreibt (äußerer Zyklus).
2. Die in die Transaktionsbearbeitung involvierten Kontrollinstanzen koordinieren die Phasenübergänge der Transaktionen. Die globale Rangordnung wird zur Auflösung lokaler Verklemmungssituationen benutzt (innerer Zyklus).

Äußerer Zyklus:

Die Erstellung der globalen Rangordnung für die Transaktionen hat durch die Kontrollinstanzen gemeinsam zu erfolgen und erfordert deren feste Kopplung. Die Koordinationsanforderung bezieht sich auf die Durchführung der Phase 1 (Einreihung) einer Transaktion. Das in 4.2. entwickelte verbesserte elementare Verfahren kann vorteilhaft zur Koordination der Einreihung der Transaktionen

in die globale Rangordnung verwendet werden; während der Abstimmungsphase werden die noch nicht eingereihten Transaktionen gesammelt. Im Zustand 3 erfolgt die Erstellung der Rangordnung für diese Transaktionen. (Die Rangordnung ist im Sinne einer Präzedenzregelung (irreflexive Halbordnung /C5/) zu verstehen.) Die in einem Zyklus erhaltene Rangordnung wird an die im vorangegangenen Zyklus erhaltene angehängt (Konkatenation der Rangordnungen).

Sobald eine Kontrollinstanz eine oder mehrere Koordinationsanforderungen vorliegen hat, initialisiert sie einen solchen Zyklus vom Zustand 1 aus. Die restlichen Kontrollinstanzen folgen ihr, sobald sie aus dem inneren Zyklus in den Zustand 1 zurückgekehrt sind. Bei Fremdinitialisierung hat der äußere Zyklus Vorrang gegenüber dem inneren; eine Eigeninitialisierung kann z.B. in Abhängigkeit von der Anzahl der auf ihre Einreihung wartenden Transaktionen erfolgen.

Innerer Zyklus:

Im Zustand 1 wird analysiert, ob in den äußeren oder in den inneren Zyklus überzugehen ist.

In den inneren Zyklus kann geschaltet werden - Übergang in den Zustand 5 -, falls eine eingereihte Transaktion (von lokaler Seite aus) einen der Phasenübergänge $1 \rightarrow 2$, $2 \rightarrow 4$ oder $4 \rightarrow 5$ beantragt. Ist lokal einer dieser Phasenübergänge möglich, benachrichtigt die Kontrollinstanz alle in die Bearbeitung der Transaktion involvierten Kontrollinstanzen unter Angabe des Transaktionsnamens und des Phasenübergangs. Die Kontrollinstanz prüft, ob die anderen Kontrollinstanzen ihre Bereitschaft angezeigt haben, d.h. ob der Phasenübergang global erlaubt ist. Liegt die Erlaubnis vor, wird der Phasenübergang vollzogen und im Bearbeitungszustand der Transaktion die Phase vermerkt. Danach kehrt die Kontrollinstanz in den Zustand 1 zurück.

Bevor für eine Transaktion T_1 der Phasenübergang $2 \rightarrow 4$ als lokal erlaubt markiert wird, muß zuerst geprüft werden, ob nicht eine Transaktion T_2 existiert, die in der globalen Rangordnung vor T_1 angesiedelt und deren Phasenübergang $2 \rightarrow 4$ lokal noch nicht erlaubt ist. In diesem Fall kann es eintreten, daß eine solche Transaktion T_2 im weiteren Verlauf ihrer Bindungsphase

eine Verdrängung von T_1 fordern könnte. Die lokale Erlaubnis für den Phasenübergang 2 → 4 kann einer Transaktion also erst erteilt werden, falls allen ihren Vorgänger-Transaktionen in der globalen Rangordnung dieser Phasenübergang lokal erlaubt wurde.

Der Übergang in den Zustand 5 erfolgt auch, wenn eine Nachricht von einer anderen Kontrollinstanz erhalten wird, die sich auf eine bereits eingereihte Transaktion bezieht. In diesem Fall wird der Phasenübergang der Transaktion für diese Kontrollinstanz als lokal erlaubt markiert und geprüft, ob der Phasenübergang global erlaubt ist und damit (lokal) vollzogen werden kann.

Bei Freigabe von gesperrten Objekten werden lokal blockierte Transaktionen wieder in die Bindungsphase überführt.

Wird der Phasenübergang 4 → 2 für eine Transaktion zugelassen - Aufhebung der Einschränkung der sukzessiven Betriebsmittelanforderung -, können globale Verklemmungssituationen entstehen, die lokal nicht als solche erkannt und damit auch nicht aufgelöst werden können. Dies erfordert eine Kooperation aller Kontrollinstanzen, zumal wenn zur Auflösung von Verklemmungssituationen Transaktionen koordiniert zurückgesetzt werden müssen. Die Behandlung einer solchen Situation kann in den äußeren Zyklus eingebettet werden, indem sie als zusätzliche Koordinationsanforderung mitberücksichtigt wird. Eine entsprechende Verfahrensregelung wird in 4.3.2. detailliert vorgestellt.

Das Verfahren gewährleistet die verklemmungsfreie Koordinierung der Transaktionsbearbeitung und die operationale Integrität der verteilten Datenbasis, falls Transaktionen nicht diskriminiert werden:

- das verbesserte elementare Verfahren im äußeren Zyklus, der eine höhere Priorität als der innere besitzt, sichert die Erstellung einer systemweit gültigen Transaktionsrangordnung, die globale Verklemmungssituationen verhindert,
- lokale Verklemmungssituationen löst der lokale Sperrmechanismus unter Berücksichtigung der globalen Rangordnung der Transaktionen,

- die Koordinierung der Phasenübergänge von Transaktionen durch mehrere Kontrollinstanzen ist durch die Regelung des Übergangs von der lokalen zur globalen Vollziehbarkeit der Phasenübergänge gesichert.

4.3.2. Zulassen, Erkennen und Beseitigen von Verklemmungen

Die Grundzüge des Verfahrens sind (vgl. 4.4.2.):

1. Eine Kontrollinstanz bearbeitet eine Transaktion in Kooperation mit den in ihre Bearbeitung miteinzubeziehenden Kontrollinstanzen (innerer Zyklus).
2. Periodisch oder bei Eintreten von Blockierungen von globalen Transaktionen durch andere globale oder lokal begrenzte Transaktionen ist eine Kooperation aller Kontrollinstanzen einzuleiten, die eine koordinierte Erkennung und Beseitigung von Verklemmungssituationen zum Ziele hat (äußerer Zyklus).

Innerer Zyklus:

Der Eintritt in den inneren Zyklus wird wie in 4.3.1. durch entsprechende Ereignisse veranlaßt. Zu koordinieren sind die Phasenübergänge 1 → 2, 2 → 4, 4 → 2, 4 → 5 für globale Transaktionen. (Die Phasenübergänge 2 → 3 bzw. 3 → 2 müssen im äußeren Zyklus behandelt werden.) Es wird wie in 4.3.1. verfahren, d.h. ein Phasenübergang wird lokal erst durchgeführt, wenn er global vollziehbar ist.

Äußerer Zyklus:

Beantragt eine Kontrollinstanz eine Durchführung des Algorithmus zur Erkennung und Beseitigung von möglicherweise existierenden Verklemmungssituationen, so müssen die anderen Kontrollinstanzen, falls aus dem inneren Zyklus nach Zustand 1 zurückgekehrt, dieser Aufforderung Folge leisten. Die Koordination gelingt mittels einer Vereinfachung des Basisprotokolls. In der Abstimmungsphase ist keine Verdrängung konkurrierender Koordinationsanforderungen zu berücksichtigen, da Eigeninitialisierungen unterschiedlicher Kontrollinstanzen sich auf ein- und dieselbe im Zustand 3 durchzuführende Aktivität beziehen. In der

Bereitschaftserklärung, die eine Kontrollinstanz als Eigeninitialisierung oder in Beantwortung einer Fremdinitialisierung allen anderen Kontrollinstanzen übermittelt, zeigt sie den anderen Kontrollinstanzen ihre lokal existierende Zustandsinformation hinsichtlich der Blockierung globaler Transaktionen an. Nach Empfang aller Bereitschaftserklärungen kennt eine Kontrollinstanz den gesamten globalen Blockierungszustand.

Im Zustand 3 führt jede Kontrollinstanz den (für jede Kontrollinstanz identischen) Algorithmus zur Erkennung und Beseitigung von Verklemmungen aus. Transaktionen, deren Betriebsmittel entzogen werden müssen, werden rückgesetzt. Danach kehren die Kontrollinstanzen geordnet in den Zustand 1 zurück.

Das Verfahren gewährleistet die verklemmungsfreie Koordinierung der Transaktionsbearbeitung und sichert die operationale Integrität der verteilten Datenbasis:

- lokal beschränkte Verklemmungssituationen werden durch den lokalen Sperrmechanismus gelöst,
- globale Verklemmungssituationen werden durch das im äußeren Zyklus angewandte vereinfachte Basisprotokoll und der höheren Priorität des äußeren Zyklus gegenüber der des inneren gelöst,
- die Koordinierung der Phasenübergänge von Transaktionen durch mehrere Kontrollinstanzen ist durch die Regelung des Übergangs von der lokalen zur globalen Vollziehbarkeit der Phasenübergänge gesichert.

Die in 4.3.1. erwähnte Strategie zur Verhinderung von Verklemmungen durch Entzug aller Betriebsmittel kann mit einer Modifikation des Verfahrens eingesetzt werden. Sobald im inneren Zyklus eine Blockierung einer Transaktion bemerkt wird, wird sie in Kooperation mit den anderen in ihre Bearbeitung involvierten Kontrollinstanzen rückgesetzt und nach einer gewissen Zeit neu gestartet. Um eine Diskriminierung einer Transaktion zu verhindern, ist diese Rücksetzung mit einer Prioritätserhöhung zu verbinden, die bei späteren Verdrängungen derart berücksichtigt wird, daß in einem äußeren Zyklus die Transaktionen niederer Priorität rückgesetzt werden und die Transaktion schließlich doch bearbeitet werden kann.

4.4. Beispiel für die Anwendung ausgewählter Verfahren

Die Datenbasis bestehe aus den Variablen a,b,c,x,y. Die Variablen a,b,c stellen Meßgrößen dar, deren Werte unabhängig voneinander verändert werden können. Als Konsistenzregeln seien festgelegt, daß x den Mittelwert von a und b enthalten soll, und y den von b und c.

Das System umfasse zwei Kontrollinstanzen, denen die Variablen als Konstituenten wie folgt zugeordnet seien:

- Kontrollinstanz 1: a,b,x
- Kontrollinstanz 2: c,y

Transaktionen dürfen die Werte der Variablen lesen oder verändern unter gleichzeitiger Berücksichtigung der Konsistenzregeln.

Sperreinheiten seien die einzelnen Konstituenten (einheitliche Granulierung); von einer Transaktion T wird gefordert, daß sie zu Beginn alle Variablen ihres Konsistenzbereiches in einer Sperranweisung S(T) mit den entsprechenden Sperrungsstufen spezifiziert (vgl. 2.2). Als Sperrungsstufen seien festgelegt:

- keine Sperrung (-),
- Lesesperrung (l),
- Veränderungssperrung (v).

Wir betrachten folgende Transaktionen T,U,V,W:

S(T) = ((a,v), (b,l), (x,v));
T verändert a und aktualisiert x, wozu b mit Lesesperrung belegt werden muß.

S(U) = ((a,l), (b,v), (c,l), (x,v), (y,v));
U verändert b und aktualisiert x und y; hierzu müssen a und c mit Lesesperrung belegt werden.

S(V) = ((b,l), (c,v), (y,v));
V verändert c und aktualisiert y unter entsprechender Sperrung von b.

S(W) = ((b,l));
W möchte lediglich den Wert von b lesen.

Die Transaktionen T,V,W können parallel ausgeführt werden, U dagegen kollidiert mit allen anderen Transaktionen. Die Transaktionen T,U und W sollen in dieser Reihenfolge bei Kontrollinstanz 1 ankommen, bei Kontrollinstanz 2 treffe Transaktion V ein.

Die Reihenfolge der Transaktionsbearbeitung richte sich nach folgenden Regeln:

- die Priorität der Kontrollinstanz 2 ist höher als die der Kontrollinstanz 1 (Verdrängungsregelung),
- die Priorität einer Transaktion innerhalb einer Kontrollinstanz sei durch ihre Position in der Eingangsreihenfolge bestimmt,
- Transaktionen, die mit Transaktionen höherer Priorität oder mit in Ausführung befindlichen Transaktionen kollidieren, werden rückgestellt (zur Verhinderung der hierdurch möglichen Diskriminierung von Transaktionen wäre dieses Prioritätsschema zu modifizieren).

Wir wählen folgende Verfahren:

- das elementare Verfahren (4.2.1) als Vertreter der Verfahren mit fester Kopplung,
- das Verfahren mit Verklemmungsbeseitigung (4.3.2) als Vertreter der Verfahren mit loser Kopplung.

4.4.1. Anwendung des elementaren Verfahrens

In jeder Kontrollinstanz ist zu jeder der Sperreinheiten (SE) folgende Sperrinformation zu führen (vgl. 2.2):

- Sperrungsstufe (SS),
- Liste der Transaktionen, die an der Sperrung beteiligt sind (L).

Wir betrachten folgende Ausprägungen der Sperrinformation:

SE	Ausprägung 1		Ausprägung 2		Ausprägung 3		Ausprägung 4	
	SS	L	SS	L	SS	L	SS	L
a	-	-	v	T	v	T	v	T
b	-	-	l	T	l	T,V	l	T,V,W
c	-	-	-	-	v	V	v	V
x	-	-	v	T	v	T	v	T
y	-	-	-	-	v	V	v	V

Ausgangspunkt ist das leere System (Ausprägung 1). Nach Eingang von T führt die Ausführung der Sperranweisung von T (Belegung von Betriebsmitteln) im Zustand 3 des Zyklus zur Ausprägung 2 der Sperrinformation bei jeder Kontrollinstanz. Danach kehren die Kontrollinstanzen in den Zustand 1 zurück, und Transaktion T befindet sich in ihrer Ausführungsphase. Diese Situation bildet den Ausgangspunkt für die in Bild 4.4 dargestellte Koordinierung des Beginns der Ausführungsphasen der Transaktionen V und W (vgl. hierzu auch Bild 3.5); U wird wegen Kollision mit T rückgestellt. Bei einer Beendigung der Transaktionen T,V und W in der Reihenfolge W,V,T und der damit verbundenen Freigabe von Sperrungen werden die Ausprägungen der Sperrinformation entsprechend koordiniert verändert. Nach Beendigung von T besitzt die Sperrinformation die Ausprägung 1; die Sperranweisung von U kann nun ausgeführt werden.

4.4.2. Anwendung des Verfahrens mit Verklemmungsbeseitigung

Jede Kontrollinstanz führt nur Sperrinformation über die ihr zugeordneten Konstituenten.
Ausgangspunkt sei das leere System. Die Eingangsreihenfolge der Transaktionen sei wie bei 1.
Bild 4.5 zeigt das Systemgeschehen bis zum Start der Ausführung von Transaktion V. Folgende Ausprägungen der Sperrinformation der Kontrollinstanzen treten dabei auf:

Kontrollinstanz 1:

SE	Ausprägung 1		Ausprägung 2		Ausprägung 3		Ausprägung 4	
	SS	L	SS	L	SS	L	SS	L
a	-	-	v	T	v	T	v	T
b	-	-	l	T	l	T,W	l	T,W,V
x	-	-	v	T	v	T	v	T

Kontrollinstanz 2:

SE	Ausprägung 1		Ausprägung 2	
	SS	L	SS	L
c	-	-	v	V
y	-	-	v	V

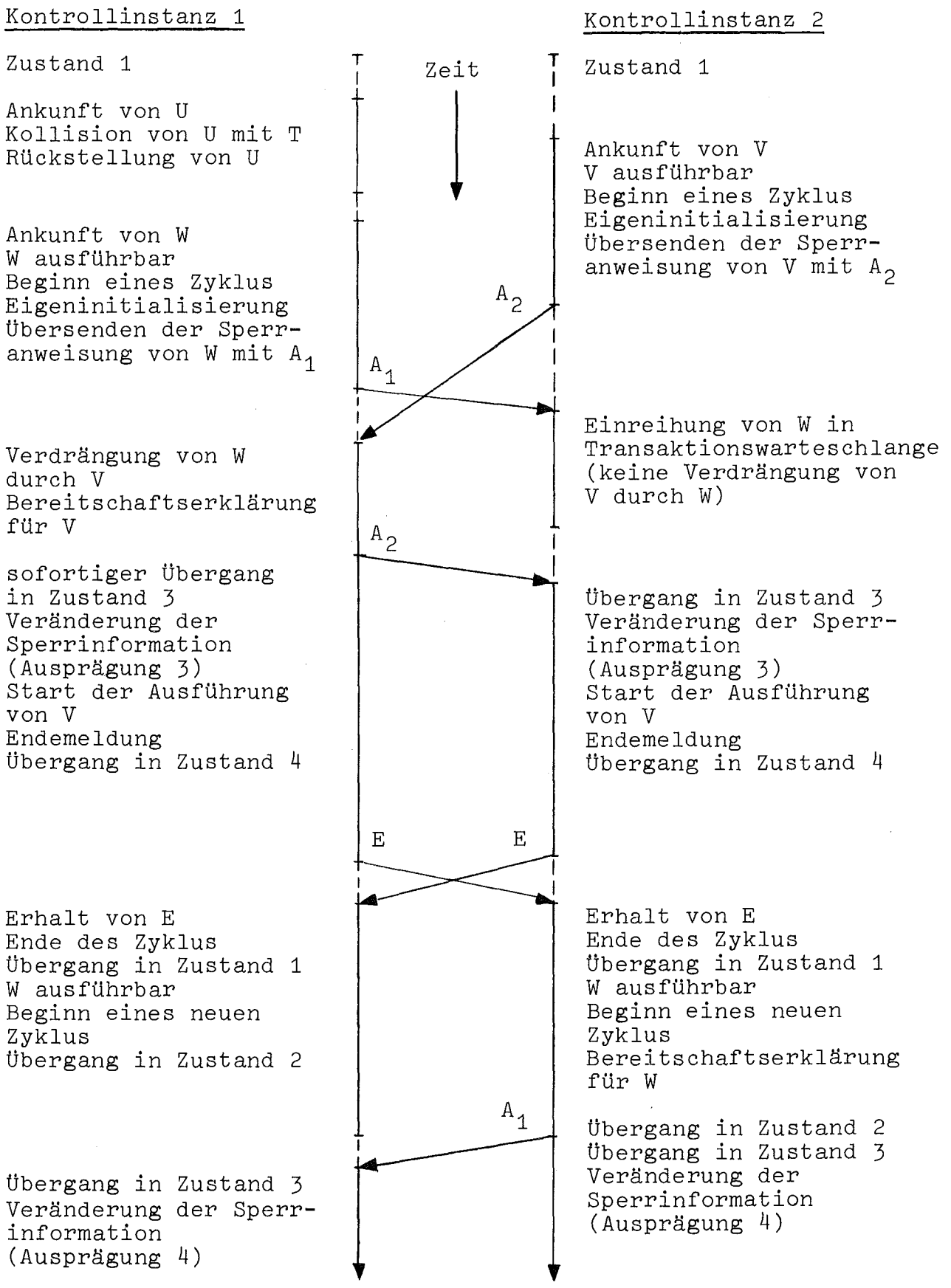


Bild 4.4: Anwendung des elementaren Verfahrens in einem System von zwei Kontrollinstanzen (---- Wartezeiten, —— Aktivzeiten der Kontrollinstanzen)

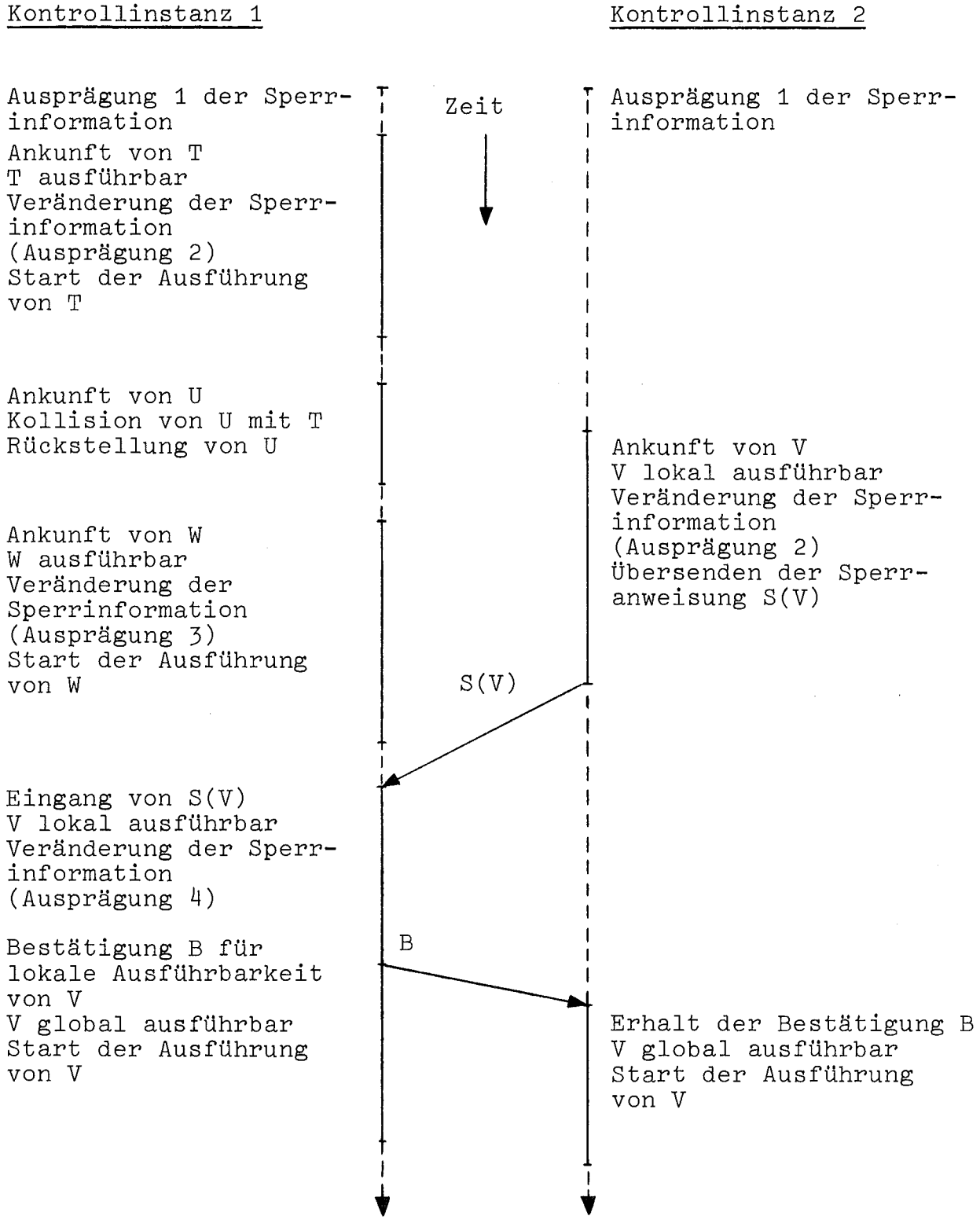


Bild 4.5: Anwendung des Verfahrens mit Verklemmungsbeseitigung in einem System von zwei Kontrollinstanzen (---- Wartezeiten, —— Aktivzeiten der Kontrollinstanzen)

Bei einer Beendigung der Transaktionen in der Reihenfolge V,W,T werden die Ausprägungen der Sperrinformation entsprechend geändert. Eine Koordinierung der Kontrollinstanzaktivitäten ist nur für die Freigabe der von V belegten Sperrungen notwendig. Nach Beendigung von T kann die Sperranweisung von U ausgeführt werden; hierzu müssen - wie bei Transaktion V - die Kontrollinstanzen kooperieren.

Beide Kontrollinstanzen befinden sich in diesem Beispiel ständig im inneren Zyklus.

Der Koordinierungsaufwand dieses Verfahrens ist gegenüber dem des elementaren Verfahrens wesentlich geringer; dies gilt jedoch nicht allgemein (siehe hierzu 4.5.).

4.5. Einsatzbereiche und Koordinierungsaufwand der Verfahren

Die erarbeiteten Verfahren unterscheiden sich

- im Umfang der zu führenden Zustandsinformation,
- in der Intensität des Nachrichtenverkehrs zwischen den Kontrollinstanzen,
- in den Aktivitäten der Kontrollinstanzen zur Transaktionskoordinierung.

Inwieweit sich diese Unterschiede auch in unterschiedlichen Belastungen der Komponenten eines verteilten DV-Systems (Rechner, Nachrichtentransportsystem) manifestieren und die Bestimmung des Verfahrens mit minimalem Koordinierungsaufwand gestatten, ist nicht nur von der Leistungsfähigkeit des verteilten DV-Systems, sondern auch von den Charakteristiken des Transaktionsprofils und der verteilten Datenbasis abhängig.

Zu den Charakteristiken des Transaktionsprofils gehören u.a.
- die Kompliziertheit der Operationen einer Transaktion,
- der Anteil lesender bzw. schreibender Operationen im Transaktionsstrom,
- der Umfang des Konsistenzbereichs der Transaktionen,
- die Ankunftsrate kollidierender Transaktionen.

Charakteristiken der verteilten Datenbasis sind z.B.:

- das verwendete Datenmodell,
- die Topologie der verteilten Datenbasis (Grad der redundanten

Realisierung bzw. strukturellen Abhängigkeit von Konstituenten, Zuordnung der Konstituenten zu Kontrollinstanzen usw.),
- der Grad der Dezentralisierung der Kontrolle,
- die Granularität der Sperreinheiten.

Die Verfahren mit fester Kopplung sind sinnvoll einsetzbar, wenn

- eine grobe Granulierung der Sperreinheiten vorliegt und damit die zu führende Zustandsinformation von geringem Umfang ist,
- die Konsistenzbereiche der Transaktionen und die Zuordnung von Konstituenten zu Kontrollinstanzen verlangen, daß in die Bearbeitung einer jeden Transaktion der Großteil der Kontrollinstanzen einzubeziehen ist.

In einer solchen Umgebung sind Verfahren mit fester Kopplung solchen mit loser Kopplung vorzuziehen; das Verfahren mit Vorsortierung der Transaktionen würde fast den doppelten Aufwand benötigen, da Vorsortierung und Transaktionsstart in getrennten Zyklen behandelt werden; bei Verfahren mit Verklemmungsbeseitigung wären globale Verklemmungssituationen sehr häufig, falls unterschiedliche Kontrollinstanzen gleichzeitig Transaktionsbearbeitungen starten.

Verfahren mit loser Kopplung sind gegenüber dem Parameter Granulierung von Sperreinheiten insensitiv. Verfahren mit Verklemmungsbeseitigung sind optimal im Vergleich zu anderen Verfahren, falls das Transaktionsprofil und die Verteilungsstruktur der Datenbasis derart sind, daß sehr viele Transaktionen lokal und ohne Koordination mit anderen Kontrollinstanzen bearbeitet werden können, d.h. globale Verklemmungssituationen sehr selten sind; dies gilt z.B.

- bei hohem Redundanzgrad der Datenbasis, hohem Leser-Anteil am Transaktionsprofil und bei geringem Umfang der Konsistenzbereiche der Transaktionen,
- oder bei niedrigem Redundanzgrad der Datenbasis und geringer struktureller Abhängigkeit der Konstituenten der Kontrollinstanzen (in diesem Fall ist die Leser - Schreiber Verteilung im Transaktionsprofil irrelevant).

Das Verfahren mit Verklemmungsverhinderung nimmt eine Mittelstellung zwischen Verfahren mit fester Kopplung und dem Verfahren mit Verklemmungsbeseitigung ein. Es empfiehlt sich

- bei hohem Redundanzgrad der Datenbasis und ausgeglichener Leser - Schreiber Verteilung im Transaktionsprofil (der Umfang der Konsistenzbereiche ist hier irrelevant),
- oder bei niedrigem Redundanzgrad der Datenbasis und einer strukturellen Abhängigkeit der Konstituenten derart, daß in die Bearbeitung einer Transaktion in der Regel zwar mehr als eine aber nicht der Großteil der Kontrollinstanzen einzubeziehen ist.

Diese grobe Analyse des qualitativen Einflusses einiger Parameter zeigt, daß jedem Verfahren ein Einsatzbereich zugeordnet werden kann, in dem es nicht nur sinnvoll einsetzbar, sondern auch den anderen Verfahren bezüglich der Minimalität des Koordinierungsaufwands überlegen ist.

Eine detailliertere qualitative und quantitative Analyse des Leistungsverhaltens der vorgestellten Verfahren bleibt zukünftigen Arbeiten vorbehalten; hierzu sind erst noch geeignete Modelle (z.B. Simulationsmodelle) oder Pilotsysteme zu entwikkeln, die die Untersuchung der vielfältigen Wechselbeziehungen der angegebenen Parameter gestatten.

4.6. Ausblick auf Erweiterungen der Verfahren

Die entwickelten Verfahren sind zu erweitern, falls die Annahmen (vgl. 2.4. und 4.1.)

- einheitliche Granulierung von Sperreinheiten,
- feste Anzahl von Sperreinheiten,
- Spezifikation des Sperrbereichs nur über Benennung der benötigten Objekte und nicht über deren Inhalt

wegfallen. Mögliche Lösungsansätze für derartige Erweiterungen sollen an dieser Stelle angedeutet werden.

Die Einführung unterschiedlicher Granulierungsebenen für Sperrungen bedeutet u.U. nur eine Erhöhung der Anzahl zu verwaltender Sperreinheiten und zu berücksichtigender Sperrungsstufen. In diesem Fall können Verfahren mit fester Kopplung

unmittelbar angewendet werden, da jede Kontrollinstanz über alle Sperrungen informiert ist. Für die Einsetzbarkeit der Verfahren mit loser Kopplung ist zu fordern, daß die Sperrinformation einer jeden Kontrollinstanz diejenigen Sperreinheiten aller übergeordneten Granulierungsebenen umfaßt, die zur Sperrung der lokal vorhandenen Datenbasisobjekte erforderlich sind.

An einer Variierung der Anzahl der Sperreinheiten beim Hinzufügen oder beim Eliminieren von Datenbasisobjekten sind i.a. alle Kontrollinstanzen zu beteiligen, da die bei jeder Kontrollinstanz zu führende Zustandsinformation über die Zuordnung Konstituente - Kontrollinstanz u.U. geändert werden muß. Das Eliminieren von Datenbasisobjekten ist einer Transaktion nur erlaubt, falls sie diese zuvor mit Veränderungssperrung belegt und damit für sich exklusiv reserviert hat. Das Hinzufügen von Datenbasisobjekten kann in den Verfahren gestattet werden, wenn die Materialisation von Phantomen ausgeschlossen ist (z.B. bei serieller Abarbeitung von inhaltsverändernden Transaktionen) und die Objekte mit der ihrer weiteren Verwendung innerhalb der Transaktion entsprechenden Sperrungsstufe belegt werden.

Dürfen Transaktionen die von ihnen zu sperrenden Betriebsmittel über deren Inhalt spezifizieren, so sind zwei Fälle zu unterscheiden:

- Die Sperrung kann ohne Inspektion der Datenbasis und nur durch Vergleich mit den in der Sperrinformation erfaßten Prädikatssperrungen erfolgen. Die Verfahren mit fester Kopplung sind unmittelbar anwendbar, weil die Sperrinformation einer jeden Kontrollinstanz alle genehmigten Prädikatssperrungen enthält und diese nur in gemeinsamer Abstimmung aller Kontrollinstanzen geändert wird. Verfahren mit loser Kopplung sind ebenfalls einsetzbar, wenn für jede Prädikatssperrung im voraus alle in die Sperrung einzubeziehenden Kontrollinstanzen bestimmt werden können.
- Zur Sperrung ist eine Inspektion des Datenbasisinhalts /C2/ erforderlich. Es können diejenigen Verfahren eingesetzt wer-

den, in denen Transaktionen sukzessive Betriebsmittelbelegungen anfordern dürfen und die Strategie der Verklemmungsbeseitigung verwendet wird. Die sukzessiven Betriebsmittelbelegungen beziehen sich nun auch auf den Sperrvorgang in der Bindungsphase. Nach Feststellung der als nächste zu belegenden Sperreinheit wird bei deren Verfügbarkeit eine entsprechende Koordinationsanforderung generiert und die Bindungsphase unterbrochen. Bei erfolgter globaler Sperrung der Sperreinheit wird die unterbrochene Bindungsphase weitergeführt, falls sie noch nicht abgeschlossen ist. Im Falle der Blockierung von Transaktionen während der Bindungsphase ist bei Freigabe der entsprechenden Sperreinheit aus einer Veränderungssperrung zu prüfen, ob die Forderungen der blockierten Transaktionen auf Belegung noch weiterbestehen oder durch die Inhaltsveränderung hinfällig geworden sind.

Abschließend ist zu bemerken, daß eine Verdrängung von in Ausführung (Phase 4) befindlichen Transaktionen durch Transaktionen höherer Priorität bei allen Verfahren als gesonderte Koordinationsanforderung zu berücksichtigen ist, da hierzu i.a. eine koordinierte Rücksetzung von Transaktionen erforderlich ist.

5. Fehlertolerante Koordinierungsmechanismen

5.1. Forderungen an fehlertolerante Koordinierungsmechanismen

In einem Rechnernetz sind wegen der inhärenten Unzuverlässigkeit von Systemkomponenten Störsituationen in Rechnern und im Nachrichtentransportsystem nicht auszuschließen. Ausfälle von Speichermedien können zum Verlust von Konstituenten der Datenbasis oder zum Ausfall von Kontrollinstanzen führen, fehlerhaftes Arbeiten der Rechner zu unkontrollierten Veränderungen des Datenbasisinhalts, Fehlfunktionen im Nachrichtentransportsystem zum Ausbleiben erwarteter Nachrichten und damit zum gegenseitigen Blockieren der Kontrollinstanzen usw. Störsituationen können die Funktionsfähigkeit und Leistungsfähigkeit des Gesamtsystems wesentlich mindern, falls das System nicht fehlertolerant ausgelegt ist.

In Anlehnung an /A4/ nennen wir ein System fehlertolerant, wenn es eingebaute Fähigkeiten besitzt, um in Störsituationen sein spezifiziertes Leistungsangebot zu reduzieren und zu einem kleineren System mit vermindertem Betriebsmittelangebot zu schrumpfen (graceful degradation). Wir verlangen im Sinne von /A4/ nur eine partielle Fehlertoleranz, da ein Ausfall von Rechnern oder von Einrichtungen des Nachrichtentransportsystems u.U. nicht automatisch, sondern nur durch Eingriff von außen behoben werden kann.

Für verteilte Datenbasen bedeutet dies, daß eine redundante Auslegung von Datenbasiskonstituenten und Kontrollinstanzen notwendige Voraussetzung für eine hohe Verfügbarkeit des Gesamtsystems ist. In Ergänzung hierzu sind die unter der Annahme eines zuverlässigen Gesamtsystems entwickelten Koordinationsprotokolle zu "fehlertoleranten" Protokollen auszubauen, die die bei Ausfall und Wiedereingliederung von Konstituenten und Kontrollinstanzen vom System zu ergreifenden Maßnahmen unterstützen.

Die Verfahren zur Lösung des Multi-Kopien-Problems mit dezentraler Kontrollstruktur von 3.1. schlagen unterschiedliche Vorgehensweisen vor, um den Ausfall und die Wiedereingliederung von Konstituenten zu behandeln. Sie setzen gemeinsam voraus, daß

- die Funktionsfähigkeit der Kontrollinstanzen durch den Aus-

fall nicht gefährdet ist,
- die Fehlerentdeckung die Aufgabe der Kontrollinstanz ist, der die ausgefallene Konstituente zugeordnet ist,
- lokal Maßnahmen zur Kopiensicherung durchgeführt werden,
- Verfälschungen von Nachrichten durch das Nachrichtentransportsystem ausgeschlossen sind.

Bei den Sortierverfahren müssen die Nachrichten des Koordinationsprotokolls von den Kontrollinstanzen quittiert werden; bei Ausbleiben einer Quittung ist die Absendung der Nachricht nach Ablauf einer vorgegebenen Frist zu wiederholen (time-out, retransmit).

Das Sortierverfahren von Johnson bietet keine weiteren Sicherungsmaßnahmen an.

Das Sortierverfahren von Thomas sieht einen zweistufigen Mechanismus vor. Im ersten Schritt informiert die Kontrollinstanz A, bei der ein Ausfall aufgetreten ist, die anderen Kontrollinstanzen darüber, daß sie einen Wiederanlauf versucht. Diese müssen die Nachricht quittieren und die Weitergabe der Anforderungen einstellen, über die A schon abgestimmt hat. Im zweiten Schritt fordert A von allen Kontrollinstanzen Informationen an, die die Anforderungen betreffen, die seit dem Ausfall akzeptiert wurden oder noch nicht akzeptiert wurden, über die jedoch A bereits abgestimmt hatte. Nach erfolgreicher Übertragung dieser Informationen können die auf Grund des Ausfalls unterbrochenen Abstimmungen fortgesetzt werden. Tritt während der Stufe 2 ein weiterer Ausfall ein - bei irgendeiner Kontrollinstanz -, muß A wieder mit Schritt 1 beginnen; erst nach erfolgreichem Abschluß des zweiten Schrittes ist A wieder abstimmungsberechtigt.

Für Verfahren mit exklusiver Sperrung wird in /M5/ ein weiterer Nachrichtentyp eingeführt, mit dem eine Kontrollinstanz A den anderen Kontrollinstanzen den Ausfall ihrer Kopie anzeigt. In der Abstimmungsphase wird diese Nachricht als Bereitschaftserklärung für den Übergang in den kritischen Abschnitt gewertet; sie veranlaßt außerdem, daß die restlichen Kontrollinstanzen die danach ausgeführten Änderungen in Journalen aufbewahren und nach Wiederanlauf von A dieser zur Wiederherstellung bereitstellen. In /M5/ wird vorausgesetzt, daß im kritischen Zustand die Kontrollinstanz,

deren Transaktion (Modifikation) sich in der Abstimmungsphase durchgesetzt hat, zuerst diese Modifikation einarbeitet, und dann den restlichen Kontrollinstanzen zur Durchführung übergibt. Fällt diese Kontrollinstanz während der kritischen Phase aus, müssen die anderen Kontrollinstanzen deren Wiederherstellung abwarten, d.h. sie sind solange blockiert.
Die aufgeführten Vorgehensweisen genügen nicht den Forderungen, die in verteilten Datenbasen bei dezentraler Kontrollstruktur an die Behandlung von Störsituationen zu stellen sind:

- Wird eine Transaktion in mehreren Rechnern gleichzeitig ausgeführt, ist bei Ausfall eines in die Transaktionsbearbeitung involvierten Rechners eine koordinierte Reaktion des Restsystems (z.B. koordiniertes Rücksetzen der Transaktion) erforderlich.
- Der Ausfall einer Kontrollinstanz (Ausfall des zugehörigen Rechners, Ausfall von Kommunikationsverbindungen zu den anderen Kontrollinstanzen) darf nicht das funktionsfähige Restsystem blockieren und damit die weitere Bearbeitung von Transaktionen verhindern.

Aus diesen Gründen ist eine sofortige Reaktion der Kontrollinstanzen bei Störsituationen in fester Kopplung erforderlich und der Behandlung von Störsituationen gegenüber der Transaktionsbearbeitung Vorrang einzuräumen, d.h. die Koordinierung der Kontrollinstanzen zur Fehlerbehandlung hat in einer der Transaktionsbearbeitung überzuordnenden Ebene zu erfolgen.

Fehlertolerante Koordinierungsmechanismen müssen also im wesentlichen folgende vom System bei Ausfall und Wiedereingliederung von Konstituenten und Kontrollinstanzen zu ergreifende Maßnahmen unterstützen:

- Erkennung von Fehlern in der Transaktionsbearbeitung,
- Sicherung der operationalen Integrität für alle Transaktionen und Datenbestände des Restsystems über den Störfall hinweg,
- Rekonfiguration aller arbeitsfähigen Komponenten zu einem funktionsfähigen Restsystem,
- Aufrechterhaltung der operationalen Integrität bei paralleler Bearbeitung der im Restsystem ablauffähigen Transaktionen,
- Sicherung der Konsistenz für die Wiedereingliederung von Konstituenten.

Die Kontrollinstanzen sind zur Erledigung dieser Maßnahmen mit zusätzlichen Komponenten und Zustandsinformationen zu versehen.

Zusammen mit den in 4.1. aufgeführten Komponenten enthält eine Kontrollinstanz nun

- Mechanismen zur Datensicherung und Rekonfiguration,
- Sperrmechanismus,
- Protokolleinheit zur Koordination der Transaktionsbearbeitung und Fehlerbehandlung.

Die Zustandsinformation umfaßt

- den Aktiv-Zustand der Kontrollinstanzen,
- den Datenbasiszustand,
  - o Aktiv-Zustand der Konstituenten
  - o Information für Datensicherung
  - o Zuordnung Kontrollinstanzen - Konstituenten
  - o Sperrinformation
- den Transaktionsbearbeitungszustand,
- den Protokollzustand.

Die fehlertoleranten Koordinierungsmechanismen müssen daher im wesentlichen erfüllen:

- die Koordination der globalen Datensicherung zur Gewährleistung der Rückkehr der Datenbasis vom inkonsistenten in den konsistenten Zustand bei Ausfall und Wiedereingliederung von Konstituenten und Kontrollinstanzen,
- die Koordination der Veränderung der Information der Kontrollinstanzen über den Aktiv-Zustand von Konstituenten und Kontrollinstanzen bei deren Ausfall und Wiedereingliederung als notwendige Voraussetzung zur Systemrekonfiguration.

Wir werden zunächst in 5.2. Maßnahmen zur Unterstützung der globalen Datensicherung als Voraussetzung zur Konsistenzsicherung erarbeiten.

In 5.3. wird der Fall behandelt, daß Fehler den Aktiv-Zustand von Datenbasiskonstituenten aber nicht die Funktionsfähigkeit der Kontrollinstanzen beeinträchtigen.

Eine Erweiterung des Protokolls zur Reaktion auf Ausfälle von Kontrollinstanzen wird in 5.4. entwickelt.

Abschließend wird in 5.5. die Wiedereingliederung von Konstituenten und Kontrollinstanzen aufgezeigt.

5.2. Globale Datensicherung

Eine globale Datensicherung setzt die Existenz von Vorsorgemaßnahmen zur Datensicherung in den Komponenten des verteilten DV-Systems voraus.

Die Datensicherung im Nachrichtentransportsystem beinhaltet die korrekte Übertragung von Information zwischen Sendern und Empfängern. Unterschiedliche Maßnahmen können getroffen werden /M1/. Zur Absicherung von Nachrichteninhalten gegenüber Veränderungen wird aus dem zu sichernden Nachrichtentext Prüfinformation, z.B. Prüfsummen, zyklische Redundanz (Prüfpolynom) /M1/, ermittelt, die dem Nachrichtentext hinzugefügt wird. Der Empfänger der Nachricht ermittelt seinerseits die Prüfinformation aus dem Nachrichtentext und vergleicht sie mit der in der Nachricht mitgeführten. Übereinstimmung bzw. Nicht-Übereinstimmung wird dem Sender durch eine positive bzw. negative Quittung mitgeteilt. Bei negativer Quittung oder bei Ausbleiben jeglicher Quittung innerhalb einer vorgegebenen Zeitschranke (time-out), wird die Übertragung durch den Sender wiederholt (retransmit), u.U. unter Benutzung von Alternativwegen, falls das Nachrichtentransportsystem solche anbietet. Kommt nach mehrmaliger Wiederholung keine für den Sender erkennbar erfolgreiche Übertragung zustande, erfolgt eine Fehlermeldung an den Auftraggeber für die Nachrichtenübertragung.

Die bereitgestellten Sicherungsverfahren /M1/ schließen mit hoher Wahrscheinlichkeit bei zustande gekommener Übertragung eine Verfälschung der Nachricht durch das Nachrichtentransportsystem aus. Ebenso werden durch Einrichtungen wie Sequenznummern /H1/ in der Regel Duplizierungen von Nachrichten durch das Nachrichtentransportsystem erkannt. Nachrichtenduplikate, die von fehlerhaften Sendern initiiert und dem Nachrichtentransportsystem zur Übertragung übergeben werden, können von diesem nicht entdeckt werden; ihr potentielles Auftreten ist in den Protokollen der Kommunikationspartner zu berücksichtigen /L1/.

Für die Datensicherung in konventionellen (zentralisierten) Da-

tenbanken werden verschiedene Vorgehensweisen vorgeschlagen (siehe /D3,L3,M1,W1/), um Störungen - wie fehlerhaftem oder unbeabsichtigtem Überschreiben von Daten oder mechanischen Fehlern in den Speichergeräten - zu begegnen.

Zur Verhinderung von Fehlern dienen Methoden, die darauf beruhen, daß das Ergebnis einer Operation unmittelbar nach ihrer Ausführung auf seine Korrektheit (besser: Glaubwürdigkeit) überprüft und im Fehlerfall die Operation sofort wiederholt wird, z.B. Paritätsbit-Vergleich oder time-out-Überwachung. Unterstützt werden solche Methoden durch Benutzung einer geeigneten Kodierung der Information, z.B. durch Anhängen von Prüfsummen.

Andere Vorsorgemaßnahmen behandeln den Fall, in dem ein Fehler eingetreten (und erkannt) ist und das System aus diesem fehlerhaften Zustand in einen als fehlerfrei angenommenen Zustand überführt werden muß, also Maßnahmen zum Wiederanlauf und Wiederherstellung (restart and recovery).

Diese Maßnahmen - siehe /L3,M1,W1/ - stützen sich im wesentlichen auf

- die Doppelführung der gesamten Datenbasis (master- und back-up-Kopien): Veränderungen der Datenbasis werden gemeinsam in beide Inkarnationen der Datenbasis eingebracht. Für umfangreiche Datenbanken wird in /L4/ ein Verfahren vorgeschlagen, das weniger speicherplatzaufwendig ist,
- die periodische Erstellung von (physischen oder logischen) Abzügen (dumps) der gesamten Datenbasis auf Sicherungsbänder als Wiederaufsetzpunkt (checkpoint),
- das Führen von Journalen während des laufenden Systembetriebs (Transaktionsjournale, Datenbasisjournale, audit trail).

Ein Verfahren für Wiederanlauf und Wiederherstellung umfaßt nach /S1/ die Festlegung

- der Prozeduren, die eindeutig zur Behandlung bestimmter Fehlertypen herangezogen werden, z.B. Verlust der Information auf einer Platte durch Fehler in der Mechanik,
- der Menge der Aktionen (z.B. das Führen eines Transaktionsjournals), die zur Durchführung der Prozeduren notwendigerweise unternommen werden müssen; die Aktionen sind entweder ereignis-

orientiert (z.B. Vermerken von Änderungen durch Transaktionen) oder kontrolliert (z.B. das Erstellen von Wiederaufsetzpunkten zu bestimmten Zeitpunkten) vorzunehmen.

Da die Aktionen u.U. sehr kostspielig sein können, wird in /S1/ vorgeschlagen, Verfahren zum Wiederanlauf und Wiederherstellung als Optimierungsaufgabe mit dem Ziel der Kostenminimierung zu formulieren; als Randbedingungen gehen z.B. Anforderungen der Aktionen bezüglich CPU-Zeit und Speicherplatz ein.

Ein Wiederanlauf- und Wiederherstellungsverfahren kann dann z.B. wie in /C3/ zusammengesetzt sein:
Zu periodischen Zeitabständen wird ein Wiederaufsetzpunkt erstellt und für die danach bearbeiteten Transaktionen ein Transaktionsjournal geführt.

Bei Auftreten eines Fehlers wird nach dessen Reparatur der "neueste" Wiederaufsetzpunkt in das System eingespielt und die in dem Transaktionsjournal abgespeicherten Transaktionen chronologisch wieder ausgeführt. Neu hinzukommende Transaktionen müssen warten und werden erst nach erfolgter Wiederherstellung bearbeitet.

In /C3/ wird für dieses Verfahren auch anhand eines Modells vorgeschlagen, wie die optimale Periode für die Erstellung von Wiederaufsetzpunkten festzulegen ist.

Diese für zentralisierte Datenbanken entwickelten Verfahren können in verteilten Datenbasen angewendet werden, falls die Kontrollstruktur zentral organisiert ist. Sie sind aber auch bei dezentralisierter Kontrollstruktur notwendig, um lokal Daten zu sichern. Man hat in verteilten Datenbasen die Möglichkeit einer erweiterten Datensicherung durch Führen von Kopien wichtiger Datenbestände in mehreren Rechnern, so daß bei Ausfall einer Kopie diese durch Übernahme des entsprechenden Datenbestandes von einem anderen Rechner wiederhergestellt werden kann. Diese Methode ist jedoch nur effizient einsetzbar, wenn bei Kopienübertragung von einem auf den anderen Rechner keine umfangreichen Datenkonversionen (vgl. /H4/ Kap. 3.) durchgeführt werden müssen. Eine lokale Datensicherung empfiehlt sich nicht nur für Teile der Datenbasis zur rascheren Wiederherstellung, sondern auch für die von der Kontrollinstanz benötigte Zustandsinformation, insbeson-

dere wenn diese Information - wie bei den Verfahren mit loser Kopplung - nicht von den anderen Kontrollinstanzen wiedergewonnen werden kann.

Für eine globale Datensicherung in verteilten Datenbasen ist die Erstellung von Wiederaufsetzpunkten und die damit verbundene Journalführung als gemeinsame Basis für Wiederanlauf und Wiederherstellung systemweit zu koordinieren. Sie stellt eine Koordinationsanforderung dar, die eine feste Kopplung aller Kontrollinstanzen erfordert. Ihre Einbettung in das Prioritätssystem aller Koordinationsanforderungen hängt von der gewählten Vorgehensweise ab (siehe weiter unten). Sie ist bei Verfahren mit loser Kopplung dem äußeren Zyklus einzugliedern; dies erfordert den Einsatz des vereinfachten Basisprotokolls im äußeren Zyklus, da unterschiedliche Koordinationsanforderungen zu berücksichtigen sind. Durch das vereinfachte Basisprotokoll wird bei geeigneter Prioritätsregelung die verklemmungsfreie Koordination der Transaktionsbearbeitung und der Aufnahme eines konsistenten Wiederaufsetzpunktes gewährleistet.

Für die Festlegung der optimalen Periode der Erstellung von Wiederaufsetzpunkten kann die in /C3/ hergeleitete Abschätzung herangezogen werden.

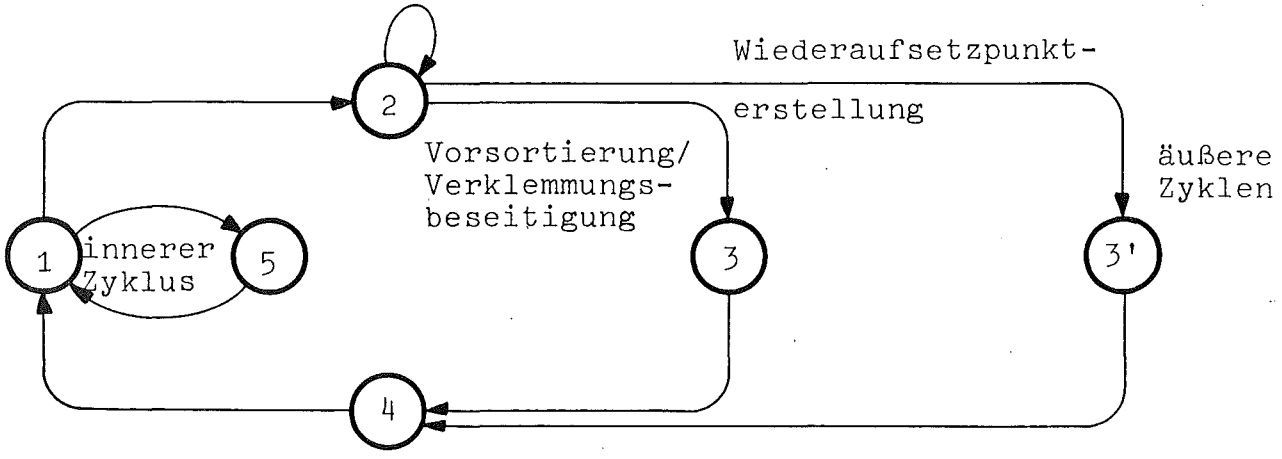


Bild 5.1: Einbettung der Erstellung von Wiederaufsetzpunkten in das Koordinationsprotokoll für Verfahren mit loser Kopplung

Zur Erstellung eines globalen Wiederaufsetzpunktes bieten sich unterschiedliche Vorgehensweisen an:

- Alle Kontrollinstanzen verharren solange in Zustand 3' bis der Wiederaufsetzpunkt erstellt ist. Während dieser Zeit dürfen keine Transaktionen auf der Datenbasis operieren, d.h. Sperreinheiten belegt halten. Dies sichert einen systemweit gültigen Wiederaufsetzpunkt und ist einzusetzen, falls systemweite Konsistenzprüfungen durchzuführen sind.
- Die Kontrollinstanzen führen die Erstellung des Wiederaufsetzpunktes in einer Ordnung seriell durch uzw. derart, daß ein Rechner diese Aktion erst beginnt, wenn sein Vorgänger sie abgeschlossen und dies ihm mitgeteilt hat. Die Gesamtheit der Kontrollinstanzen koordiniert sich nur über den Start und das Ende einer solchen Phase (zwei Koordinationszyklen im äußeren Zyklus), ansonsten teilt nur der Vorgänger seinem Nachfolger (im inneren Zyklus) mit, daß er die Aktion beendet hat. Während dieser Phase dürfen reine Leser-Transaktionen von denjenigen Kontrollinstanzen bearbeitet werden, die momentan nicht einen Wiederaufsetzpunkt aufnehmen. Eine solche Phase darf nur gestartet werden, wenn vorher eventuell bestehende Verklemmungen aufgelöst worden sind. Bei zeitaufwendigen Wiederaufsetzpunkten bietet dies gegenüber dem ersten Verfahren eine höhere Auslastung des Gesamtsystems bezüglich der Transaktionsbearbeitung.
- Als weitere Alternative bleibt die Erstellung eines Wiederaufsetzpunktes durch eine Kontrollinstanz A unabhängig von einer solchen Aktion anderer Kontrollinstanzen. Beginn und Ende dieser Aktion werden im äußeren Zyklus systemweit koordiniert. Die restlichen Kontrollinstanzen können in der Zwischenzeit auch datenbasisverändernde Transaktionen bearbeiten, sofern sichergestellt ist, daß deren Konsistenzbereiche nicht Konstituente von A enthalten. Hierdurch ist ebenfalls ein systemweit gültiger Wiederaufsetzpunkt gewährleistet. Dieses Verfahren empfiehlt sich insbesondere, wenn z.B. in einem Rechnernetz die optimalen Perioden der Erstellung von Wiederaufsetzpunkten für die einzelnen Rechner signifikant differieren.

Lokale und globale Maßnahmen zur Datensicherung sind nur notwendige Voraussetzung zur Sicherung der operationalen Integrität in verteilten Datenbasen. Sie sind um Verfahren zu ergänzen, die bei Ausfall und Wiedereingliederung von Konstituenten und

Kontrollinstanzen den Übergang in ein funktionsfähiges System zur Transaktionsbearbeitung gewährleisten.

5.3. Ausfall von Konstituenten

Fehler sollen nur den Ausfall von Konstituenten bedingen, aber nicht die Funktionsfähigkeit der Kontrollinstanzen beeinträchtigen. Sie werden durch das lokale Betriebssystem erkannt und der Kontrollinstanz gemeldet, die daraufhin eine entsprechende Kooperation der Kontrollinstanzen zur Fehlerbehandlung einleitet. Wir setzen zudem voraus, daß Nachrichteninhalte durch das Nachrichtentransportsystem nicht verfälscht werden.

Es ist zu fordern, daß jede Kontrollinstanz alle Konsistenzregeln kennt, denn mit Ausfall einer Konstituente muß der gesamte zugehörige Konsistenzbereich gegenüber Veränderungen geschützt werden, falls keine Kopie der ausgefallenen Konstituente bei einer anderen Kontrollinstanz existiert.

Ausgangspunkt ist die Beschreibung der Protokolleinheit für das Protokoll zur Transaktionsbearbeitung bei zuverlässigem Gesamtsystem durch das 7-Tupel $(S, \overline{I}_{in}, \overline{I}_{ex}, \overline{O}_{in}, \overline{O}_{ex}, \overline{M}, \overline{N})$.

Die erweiterte Protokolleinheit sei durch das 7-Tupel $(Z, I_{in}, I_{ex}, O_{in}, O_{ex}, M, N)$ bezeichnet.
Wegen der strikten Trennung der beiden Koordinierungsebenen für Transaktionsbearbeitung und Fehlerbehandlung sowie der Erfordernis des unmittelbaren Umschaltens im Fehlerfall von der Ebene der Transaktionsbearbeitung auf die Ebene der Fehlerbehandlung wird die Zustandsmenge Z als cartesisches Produkt $Z = R \times S$ definiert, worin R und S folgende Zustandsmengen bedeuten:

S = Zustandsmenge der Protokolleinheit für die Transaktionsbearbeitung im Fall eines zuverlässigen (störungsfreien) Systems,

R = Zustandsmenge der Protokolleinheit für die Behandlung des Ausfalls von Konstituenten.

Für ein System mit n Kontrollinstanzen ist

$R = \{1,2,3,4,5\}$

$I_{in} = \overline{I}_{in} \cup \{F_1,\ldots,F_n,W,X\}$

$I_{ex} = \overline{I}_{ex} \cup \{w,x,f\}$

$O_{in} = I_{in}$

$O_{ex} = \bar{O}_{ex}$

M: $Z \times (I_{in} \cup I_{ex}) \rightarrow Z$

N: $Z \times (I_{in} \cup I_{ex}) \rightarrow O_{in} \times O_{ex}$

Die Nachrichten F_i, $i = 1(1)n$, beinhalten die Bereitschaftserklärungen einer Kontrollinstanz zur Behandlung des von der Kontrollinstanz i, i=1(1)n, akzeptierten Fehlers und W bzw. X die Beendigung der Aktivität einer Kontrollinstanz im Zustand $(3,s) \in R \times S$ mit s beliebig aus S; w bzw. x sind die W bzw. X initiierenden, kontrollinstanzprivaten Anzeigen (genauere Erklärung folgt); f repräsentiert Fehlernachrichten, die zu einer Eigeninitialisierung eines Fehlerbehandlungszyklus führen.

Die Einschränkung der Zustandsübergangsfunktion M auf

$$(\{1\} \times S) \times (\bar{I}_{in} \cup \bar{I}_{ex}) \rightarrow (\{1\} \times S)$$

entspricht der Zustandsübergangsfunktion M des Protokolls für die Transaktionsverarbeitung in einem zuverlässigen System; die Einschränkung von M auf

$$(R \times \{s\}) \times ((I_{in} - \bar{I}_{in}) \cup (I_{ex} - \bar{I}_{ex})) \rightarrow (R \times \{s\})$$

für beliebiges $s \in S$ entspricht der Erweiterung der Protokolleinheit zur Fehlerbehandlung; diese Erweiterung ist in Bild 5.2 dargestellt.

Die Gesamtheit der Zustandsübergänge zeigt schematisch Bild 5.3 für ein auf fester Kopplung basierendes Verfahren zur Transaktionsbearbeitung.

Die grundlegende Vorgehensweise beruht auf dem in 3.2. vorgestellten Basisprotokoll.

Zustand (1,s), $s \in S$:
Im Zustand (1,s), $s \in S$ beliebig, der Koordinierung der Transaktionsbearbeitung werde durch eine Kontrollinstanz ein Fehler erkannt. Dies führt zur Initialisierung eines Fehlerbehandlungszyklus durch Übersenden einer Nachricht vom Typ F an die restlichen n-1 Kontrollinstanzen unter Spezifizierung des Fehlers (Namen der ausgefallenen Konstituenten der Datenbasis usw.).

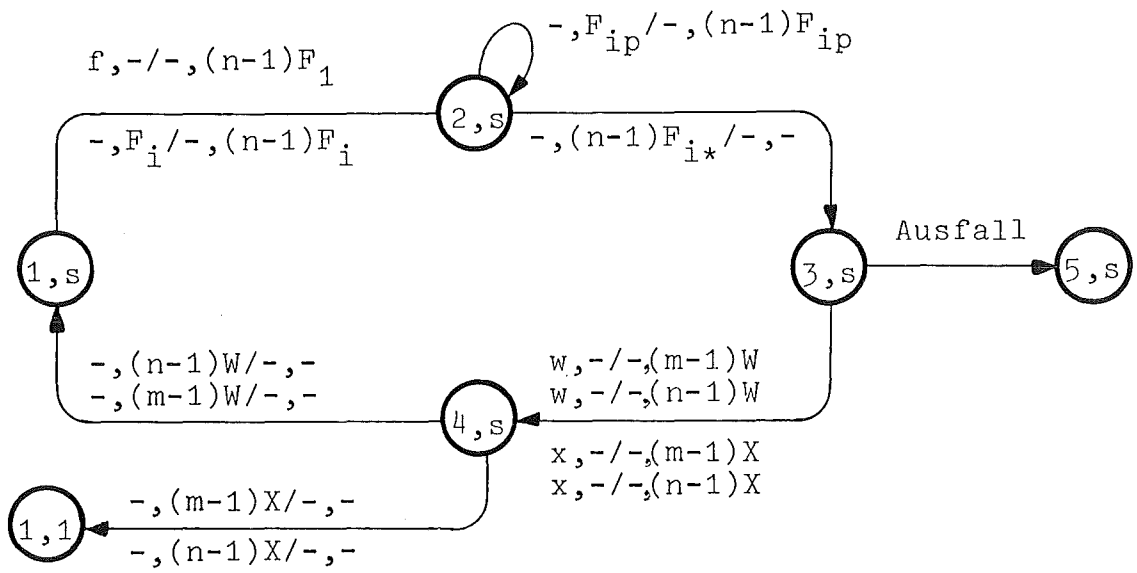


Bild 5.2: Erweiterung des Protokolls für Kontrollinstanz 1 zur Behandlung des Ausfalls von Konstituenten

Basiert das Verfahren für die Transaktionsbearbeitung auf loser Kopplung der Kontrollinstanzen, so sind zusätzlich die Transaktionen anzugeben, die von mehreren Rechnern gleichzeitig bearbeitet werden und von dem Ausfall unmittelbar betroffen sind. Die vom Fehler betroffenen Transaktionen werden in ihrer Bearbeitung unterbrochen. Die Kontrollinstanzen unterscheiden sich i.a. (insbesondere bei Verfahren mit loser Kopplung) bezüglich ihres Zustandes $s \in S$.

Zustand (2,s), $s \in S$:
Fehlerzyklen können gleichzeitig von mehreren Kontrollinstanzen initialisiert sein; im Abstimmungszyklus ist daher für die Koordinierung der Kontrollinstanzen hinsichtlich der im Zustand (3,s) gemeinsam zu behandelnden Fehler zu sorgen. Eine konfliktlösende Prioritätsregelung für Fehler ist vorzusehen, die in Verbindung mit der Verdrängungsregelung wie im Basisprotokoll die Koordinierung der Aktivitäten der Kontrollinstanzen gewährleistet.

Eine Akkumulation aller während einer Abstimmungsphase eigeninitialisierten Fehlermeldungen führt - wie beim verbesserten elementaren Verfahren in 4.2. - zu einer Verbesserung des Verfahrens; die durch die Verdrängungsregelung vorgeschriebene Quittierung durch Bereitschaftserklärungen, in denen eine Kontroll-

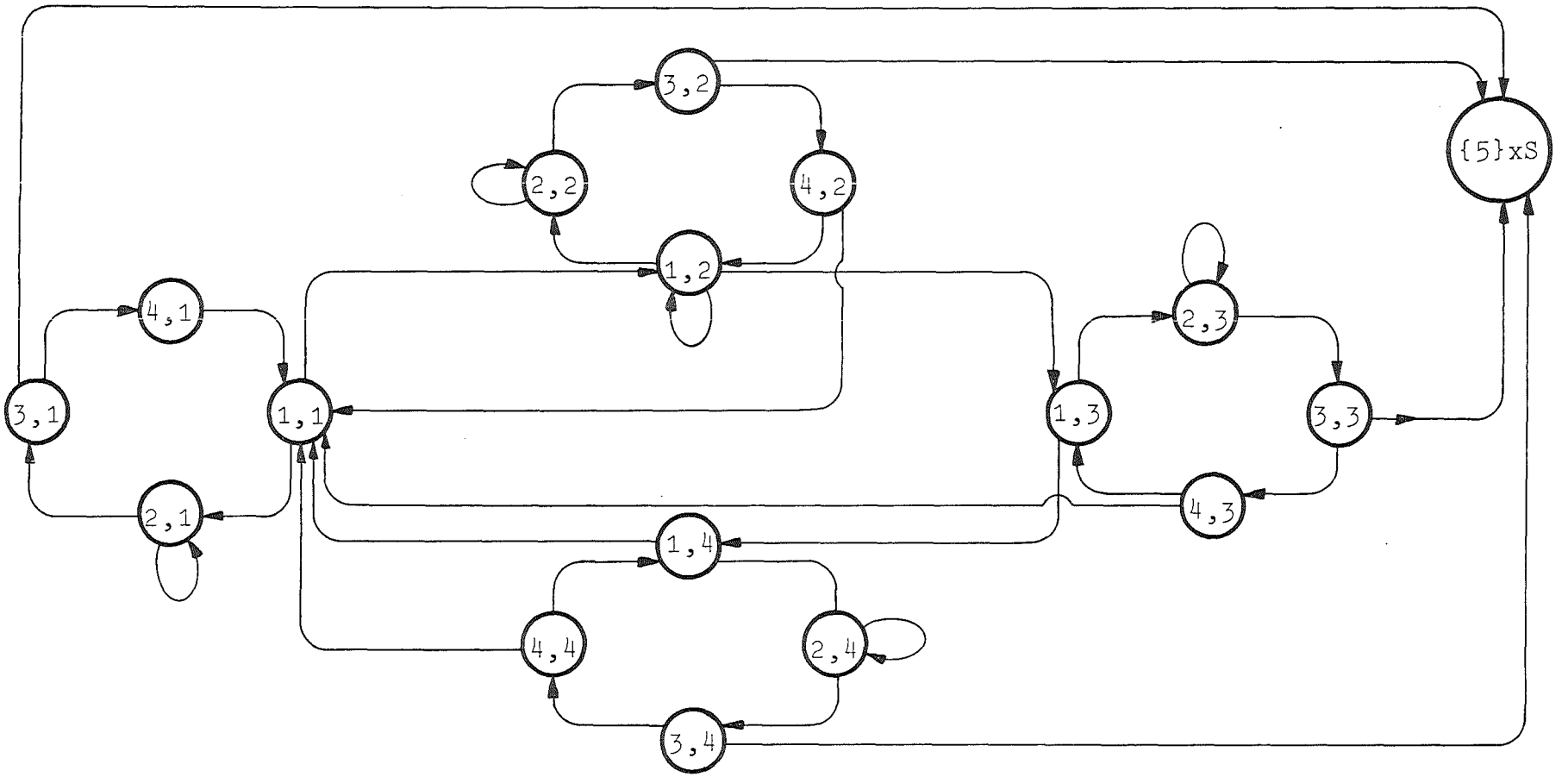


Bild 5.3: Schematische Darstellung der Zustandsübergänge einer Protokolleinheit zur Koordinierung der Transaktionsbearbeitung und des Ausfalls von Datenbasiskonstituenten mit einem auf fester Kopplung basierenden Verfahren

instanz die ihr als initialisiert bekannten Fehlermeldungen allen Kontrollinstanzen mitteilt, gewährleistet, daß alle Kontrollinstanzen bei Übergang in den Zustand (3,s) die gleiche Menge von Fehlermeldungen vorliegen haben.

Zustand (3,s), s ε S:
Nach Übergang in den Zustand (3,s) wird die Behandlung des Fehlers - bzw. (bei Akkumulation) der Fehler in der durch die Prioritätsvorschrift geregelten Reihenfolge - eingeleitet.

Die Aktivitäten der Kontrollinstanzen zur Fehlerbehandlung hängen von der Art des Fehlers sowie von den existierenden Vorsorgemaßnahmen der Datensicherung ab.

Ist eine automatische Wiederherstellung der Funktionsfähigkeit der ausgefallenen Konstituenten möglich, wird diese sofort durchgeführt, falls sie mit geringem Zeitaufwand (ein Maß für "gering" läßt sich an dem mittleren Durchsatz an Transaktionen im Gesamtsystem orientieren) verbunden ist. Würde eine Wiederherstellung die weitere Transaktionsbearbeitung durch das Restsystem unzulässig lange blockieren, so werden die ausgefallenen Konstituenten als nicht aktiv markiert und Transaktionen, die den Inhalt der mit ihnen assoziierten Objekte (in Kopien dieser Konstituenten) ändern wollen, rückgesetzt oder nicht zur Bearbeitung zugelassen. Die Kontrollinstanz, der die ausgefallenen Konstituenten zugeordnet sind, stellt deren Funktionsfähigkeit alleine (bei lokaler Kopiensicherung) oder in Zusammenarbeit mit den anderen Kontrollinstanzen, bei der diese Konstituenten als Kopie existieren, wieder her. Danach beantragt sie in einer gesonderten Koordinationsanforderung die Wiedereingliederung der reparierten Konstituenten.

Ist die Wiederherstellung nicht ohne manuellen Eingriff von außen durchführbar und kann eine Kontrollinstanz bzw. eine Menge von Kontrollinstanzen an einer weiteren Transaktionsbearbeitung nicht partizipieren, wird sie aus dem System ausgegliedert. Eine ausgegliederte Kontrollinstanz geht in den Ausfallzustand (5,s) über, aus dem sie nach erfolgreicher Reparatur ihre Reintegration beantragen muß. Das funktionsfähige Restsystem von m Kontrollinstanzen (m<n) markiert diese Kontrollinstanzen als ausgefallen und schließt sie von der weiteren Kooperation der Kontrollinstanzen aus (siehe 5.4.).

Zustand (4,s):
Nach Beendigung der Aktivitäten im Zustand (3,s) wird die Nachricht W an die als nicht ausgefallen bekannten Kontrollinstanzen ausgesandt, falls die durch den Fehlerbehandlungszyklus unterbrochene Transaktionskoordinierung fortgesetzt werden kann.
X ist auszusenden, falls eine unterbrochene Koordinierung der Transaktionsbearbeitung wieder neu begonnen werden muß, d.h. Übergang vom Zustand (4,s) in den Zustand (1,1). Die Rückkehr in den Zustand (1,s) unter W bzw. in (1,1) unter X durch das verbleibende funktionsfähige System erfolgt - wie im Basisprotokoll - geordnet.

Liegen weitere unbearbeitete Fehlermeldungen vor, ist ein neuer Fehlerbehandlungszyklus einzuleiten.

Der Nachweis der Korrektheit läßt sich auf den des Basisprotokolls zurückführen. Hierzu ist das in 3.2.2. verwendete formale Modell zu modifizieren. Wir berücksichtigen nicht die Ausgliederung von Kontrollinstanzen (siehe hierzu 5.4.).

In Anlehnung an 3.2.2. können wir den Zustand der Protokolleinheit von Kontrollinstanz i, i = 1(1)n, während eines Zyklus durch das geordnete Doppelpaar (r_iu_i, s_it_i) beschreiben mit s_it_i als dem Zustand der Protokolleinheit bezüglich der Transaktionsbearbeitung und r_iu_i als dem Zustand der Protokolleinheit bezüglich der Fehlerbehandlung mit $r_i \in \{a_i, b_i, c_i, d_i\}$, die in dieser Reihenfolge den Zuständen 1-4 von R entsprechen, und $u_i \in \{F_1,..,F_n, W, X, O\}$ (O = Leernachricht), der Menge der möglichen Nachrichten an andere Kontrollinstanzen. Die möglichen Zustände einer Kontrollinstanz ergeben sich aus der Kombination der möglichen Zustände s_it_i und r_iu_i; erstere sind durch das Protokoll der Transaktionsbearbeitung vorgegeben, letztere umfassen:

(a_i, O), $\{(b_i, F_j) : j = 1(1)n\}$, $\{c_i, F_j) : j = 1(1)n\}$,
(d_i, W) und (d_i, X).

Der Gesamtzustand eines Systems von n Kontrollinstanzen wird zu einem Zeitpunkt des Zyklus durch das geordnete n-Tupel

$$((r_1u_1, s_1t_1),\ldots,(r_nu_n, s_nt_n))$$

der möglichen Zustände der Kontrollinstanzen i, i = 1(1)n, beschrieben. Die Prioritätsfestlegung für die Kontrollinstanzen sei wie in 3.2.2. Wir setzen voraus, daß Bereitschaftserklärungen, die in einem Zyklus gegeben worden sind, in nachfolgenden Zyklen nicht mehr verwendet werden dürfen; dies gilt insbesondere für die Fehlerbehandlung.

Ein System von n Kontrollinstanzen kann unter diesen Voraussetzungen durch folgendes <n-1, n-1> TIL System G = (Σ, P, w, g) beschrieben werden mit:

Σ = {($r_i u_i$, $s_i t_i$) : i = 1(1)n und ($r_i u_i$, $s_i t_i$) ist möglicher Zustand von Kontrollinstanz i}

w = $(a_1 0, a_1 0)(a_2 0, a_2 0) \ldots (a_n 0, a_n 0)$

g = wie in 3.2.2.

P = $\{P_1, P_2\}$

Die Tabelle P_1 entspricht dem Protokoll für die Transaktionsbearbeitung - wir setzen hierfür zur Veranschaulichung das vereinfachte Basisprotokoll von 3.2.2. voraus - und P_2 dem Protokoll für die Fehlerbehandlung. Um diese Separierung vereinfacht behandeln zu können, wird im folgenden die Vollständigkeitsbedingung für die Tabellen (siehe 3.2.2.) nicht mehr gefordert (zu ihrer Beibehaltung wären zusätzliche zustandsbewahrende Produktionen einzuführen).

Die in Tabelle P_1 aufzunehmenden Produktionen sind:

1. die Produktionen des Basisprotokolls, wobei an der $r_i u_i$ repräsentierenden Stelle der Elemente von Σ nur die Symbolkombinationen $a_i 0$, i = 1(1)n, auftreten dürfen; z.B. die Produktion $a_i 0 \rightarrow a_i 0$ der Produktionsklasse 1 des Basisprotokolls hat in P_1 die Form $(a_i 0, a_i 0) \rightarrow (a_i 0, a_i 0)$.

2. zusätzlich die unter 1. genannten Produktionen mit modifizierter rechter Seite zur Anzeige der Entstehung von Fehlern: innerhalb einer Produktionsklasse wird für jedes i = 1(1)n die $r_i u_i$ repräsentierende Stelle durch $b_i F_i$ ersetzt; z.B. für die Produktion $(a_i 0, a_i 0) \rightarrow (a_i 0, a_i 0)$ ist zusätzlich die Produktion $(a_i 0, a_i 0) \rightarrow (b_i F_i, a_i 0)$ aufzunehmen.

In die Tabelle P_2 sind für alle die möglichen Zustände $s_i t_i$ darstellenden Elemente von Σ aufzunehmen

1. die Produktionen, deren Projektion bzgl. der $r_i u_i$ repräsentierenden Stellen der Elemente von Σ einer der Regeln der Produktionsklassen 1-7 des Basisprotokolls von 3.2.2. entspricht, wobei jedoch A_i durch F_i zu ersetzen ist; z.B. hat eine Produktion der Produktionsklasse 5 folgendes Aussehen für $b_i A_k$ an der Stelle $s_i t_i$: $(b_i F_j, b_i A_k) \rightarrow (b_i F_j, b_i A_k)$.

2. die Produktionen der Produktionsklassen 8-10, deren Projektion bzgl. der $r_i u_i$ repräsentierenden Stellen der Elemente von Σ die Übergänge $c_i F_j \rightarrow d_i W$, $c_i F_j \rightarrow d_i X$, $d_i X \rightarrow d_i X$, $d_i W \rightarrow d_i W$ oder $d_i W \rightarrow a_i O$ widerspiegeln.

3. die Produktionen der Produktionsklasse 10, deren Projektion bzgl. der $r_i u_i$ repräsentierenden Stellen der Elemente von Σ den Übergang $d_i X \rightarrow a_i O$ und bzgl. $s_i t_i$ den Übergang $(s_i t_i) \rightarrow a_i O$ enthalten.

Die Zusatzvorschrift (siehe 3.2.2.) für die Anwendung der Produktionen der Klassen 6 und 10 ist für die Tabelle P_2 sinngemäß zu übernehmen.

P_1 enthält also nur Produktionen, die das Protokoll für die Transaktionsbearbeitung, aber nicht für die Fehlerbehandlung regeln; umgekehrt umfaßt P_2 nur Produktionen des Protokolls für die Fehlerbehandlung und keine für die Transaktionsbearbeitung. Da zur Durchführung einer direkten Ableitung die anzuwendenden Produktionen alle aus derselben Tabelle stammen müssen, gewährleistet obige Konstruktion, daß bei Abwesenheit bzw. Anwesenheit von Fehlern P_1 bzw. P_2 gewählt und damit das gewünschte Protokoll zum Einsatz kommt.

Unter Voraussetzung der Funktionsfähigkeit der Kontrollinstanzen und der Nichtverfälschung von Nachrichten durch das Nachrichtentransportsystem erklärt sich die Korrektheit des Gesamtprotokolls wegen der Separierung der Koordinierungsebenen zur Transaktionsbearbeitung und Fehlerbehandlung aus

- der Korrektheit der beiden Protokolle in den beiden Koordinierungsebenen,

- der Übergangsregelung zwischen den Koordinierungsebenen.

Die strikte Trennung der Koordinierungsebenen spiegelt sich in der Abgrenzung des Inhalts der Tabellen P_1 und P_2 wieder.

Die Korrektheit des Protokolls für die Transaktionsbearbeitung ist durch die Ausführungen in Kap. 4 für zuverlässige Systeme sichergestellt. Das Protokoll gewährleistet die verklemmungsfreie Koordination der parallelen Bearbeitung von Transaktionen und die operationale Integrität der verteilten Datenbasis nach der erfolgreichen Rekonfiguration des Systems zu einem funktionsfähigen Restsystem.

Bei Kenntnis über einen Ausfall von Konstituenten wechseln alle Kontrollinstanzen unmittelbar in den Fehlerbehandlungszyklus über, sofern sie im Transaktionsbearbeitungszyklus sind. Der Fehlerbehandlungszyklus ist durch Koordinationsanforderungen bzgl. Transaktionsbearbeitungen nicht unterbrechbar.

Die Produktionen der Klassen 1-6 von Tabelle P_2 für den Fehlerbehandlungszyklus entsprechen den Festlegungen des vereinfachten Basisprotokolls. Sie gewährleisten nach 3.2.2., daß die Kontrollinstanzen verklemmungsfrei in den Zustand 3 des Fehlerbehandlungszyklus übergehen und, darin befindlich, konsistente Information über die zu behandelnden Ausfälle von Konstituenten besitzen.

Von der in den Fehlernachrichten mitgeführten Information über den Bearbeitungszustand der Transaktionen ist zu fordern, daß sie die Konsistenz der Aktivitäten der Kontrollinstanzen in diesem Zustand 3 zur Rekonfiguration des Systems zu einem funktionsfähigen zuverlässigen Restsystems und zur Sicherung der Integrität der Datenbasis und der im Restsystem ausführbaren Transaktionen über den Störfall hinweg gewährleistet. Bei Konsistenz der Aktivitäten senden alle Kontrollinstanzen Nachrichten vom gleichen Typ (entweder vom Typ X oder vom Typ W) aus; in diesem Fall sichern die Produktionen der Klassen 8-10, daß die Kontrollinstanzen geordnet in einen Zustand übergehen, von dem aus eine verklemmungsfreie Transaktionsbearbeitung oder eine weitere Fehlerbehandlung möglich ist.

Senden nicht alle Kontrollinstanzen nach Verlassen des Zustands 3 Nachrichten des gleichen Typs aus, so wird ein Ausfall von Kontrollinstanzen angenommen.

5.4. Ausfall von Kontrollinstanzen

5.4.1. Vereinfachtes Verfahren

Der Ausfall einer Kontrollinstanz kann durch Ausfall des zugehörigen Rechners oder von Einrichtungen des Nachrichtentransportsystems bewirkt sein. Um eine unendlich lange Blockierung der Kooperation der Kontrollinstanzen zu verhindern, ist eine Zeitüberwachung (time-out-Mechanismus) für die Aktionen aller Kontrollinstanzen durchzuführen.

Wir setzen zunächst voraus, daß

- der Ausfall von Einrichtungen des Nachrichtentransportsystems keine Zerlegung des verteilten DV-Systems in verschiedene isolierte, funktionsfähige Teilsysteme impliziert,
- Nachrichten durch das Nachrichtentransportsystem nicht verfälscht werden,
- Kontrollinstanzen fehlerlos arbeiten.

Unter diesen Prämissen wird der Ausfall einer Kontrollinstanz durch die restlichen Kontrollinstanzen "erkannt", falls erwartete Nachrichten innerhalb eines vorgegebenen Zeitintervalls ausbleiben.

Die Behandlung eines Ausfalls von Kontrollinstanzen aus den o.a. Gründen muß höchste Priorität besitzen und erfordert eine sofortige koordinierte Reaktion aller restlichen aktiven Kontrollinstanzen in einer der Behandlung des Ausfalls von Konstituenten überzuordnenden Ebene. Als Grundlage verwenden wir das vereinfachte Basisprotokoll von 3.2.

Die Erweiterung der in 5.3. definierten Protokolleinheit

$$(Z, I_{in}, I_{ex}, O_{in}, O_{ex}, M, N)$$

bezieht sich auf

- die Zustandsmenge $Z = R \times S$, der die Zustandsmenge Q für die Behandlung des Ausfalls von Kontrollinstanzen hinzuzufügen ist: $Z = Q \times R \times S$ mit $Q = R = \{1,2,3,4,5\}$,
- die Hinzunahme von Nachricht f in O_{ex},
- die Definitions- und Werte-Bereiche der Funktionen M und N (siehe Bild 5.4).

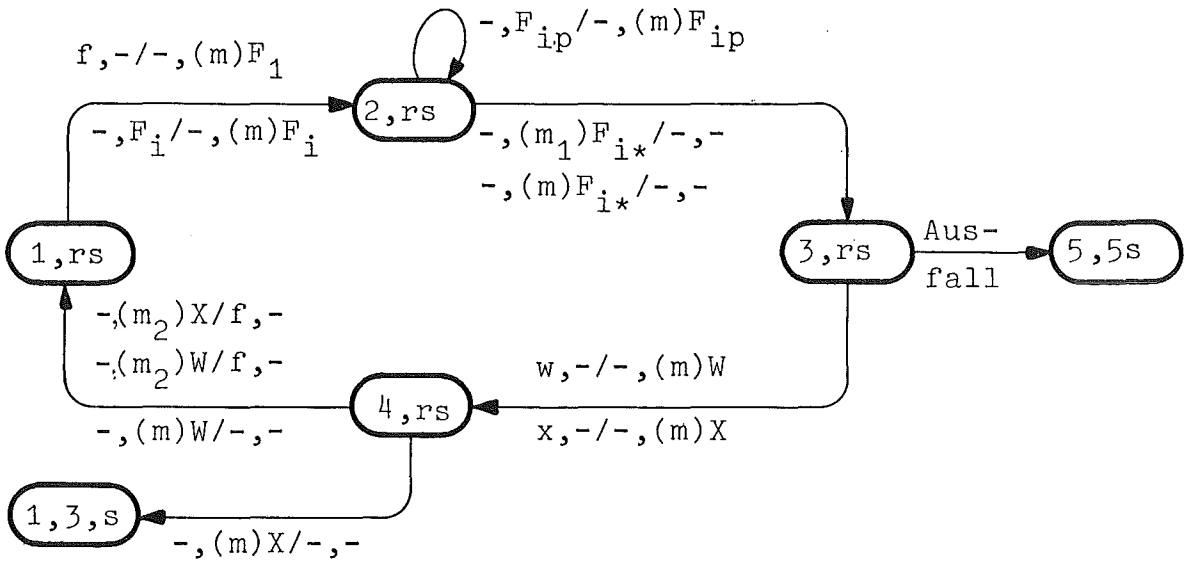


Bild 5.4: Erweiterung des Protokolls für Kontrollinstanz 1 zur Reaktion auf Ausfälle von Kontrollinstanzen

In den Nachrichten, deren Typen für die Behandlung von Ausfällen von sowohl Konstituenten als auch Kontrollinstanzen gleich sind (F, W, X usw.), ist ihr Bezug zu spezifizieren, um Mißdeutungen zu vermeiden; auf eine zusätzliche Unterscheidung wurde hier aus Vereinfachungsgründen verzichtet, da aus dem Kontext diese Eindeutigkeit hervorgeht.

Jede Kontrollinstanz i, i = 1(1)n, führt in einem Vektor z_i Information über den Aktiv-Zustand aller Kontrollinstanzen; für i, j = 1(1)n ist

$$z_{ij} = \begin{cases} 1 & \text{Kontrollinstanz i betrachtet Kontrollinstanz j als aktiv, d.h. in mögliche Kooperationen miteinzubeziehend} \\ 0 & \text{sonst} \end{cases}$$

Die Arbeitsweise der Kontrollinstanzen ist wie folgt:

Zustand (1,r,s), r ε R, s ε S:
Im Zustand (1, rs), rs wird abkürzend für (r, s) ε R x S benutzt, r ε R und s ε S beliebig, diagnostiziere eine Kontrollinstanz den Ausfall einer anderen Kontrollinstanz. Sie leitet daraufhin einen Koordinierungszyklus ein, indem sie an die restlichen m gemäß Zustandsinformation noch als aktiv bekannten Kontrollinstanzen eine entsprechende Nachricht vom Typ F sendet, in der sie neben den aus ihrer Sicht betroffenen Transaktionen den Aktiv-Zustand

angibt, den sie als neuen Aktiv-Zustand betrachtet.

Zustand (2,r,s) r ε R, s ε S:
Bei Initialisierung derselben Änderung des Aktiv-Zustands durch unterschiedliche Kontrollinstanzen muß eine Kontrollinstanz die eingehenden Meldungen für diese Fehlersituation nicht bestätigen, falls sie ihrerseits schon ihre Bereitschaft gegenüber den anderen erklärt hat. Fehlermeldungen über Ausfälle unterschiedlicher Kontrollinstanzen können initialisiert sein; es ist ebenfalls eine konfliktlösende Prioritätsregelung für die Fehlermeldungen vorzusehen; wir benutzen die des Basisprotokolls, die sich an einer linearen Anordnung der Kontrollinstanzen orientiert.

Unter Beachtung der Verdrängungsregelung wird in der Abstimmungsphase wie beim erweiterten elementaren Verfahren mit fester Kopplung (4.2) eine Akkumulation der Koordinationsanforderungen durchgeführt, die durch folgendes Beispiel erläutert werden soll:
Es seien die Kontrollinstanzen $K_1,\ldots,K_5$ gegeben; K_5 meldet einen Ausfall von K_1 mit F_5: (0,1,1,1,1); K_4 meldet einen Ausfall von K_2 mit F_4: (1,0,1,1,1). K_3 erhält F_5 und F_4 und sendet nach deren Akkumulation eine Bereitschaftserklärung von Typ F_5 mit (0,0,1,1,1) an die restlichen Kontrollinstanzen ab. Bei K_4 führt F_5 zur Verdrängung und zur Aussendung von Bereitschaftserklärungen des Typs F_5 mit (0,0,1,1,1). Hätte K_3 F_5 zuerst erhalten und protokollgerecht mit F_5: (0,1,1,1,1) beantwortet, so würde sie erst nach Eingang der Bereitschaftserklärung F_5 von K_4 den Ausfall von K_2 registrieren. Nach Eingang aller erwarteten Bereitschaftserklärungen für die Fehlermeldung des Typs F_{i*}, dessen Bearbeitung die Kontrollinstanz ebenfalls akzeptiert und dies den anderen mitgeteilt hat, wird in den Zustand (3, rs) übergegangen.

Zustand (3,r,s), r ε R, s ε S:
Aus den eingegangenen Bereitschaftserklärungen des Typs F_{i*} ermittelt die Kontrollinstanz den als vorläufig neu zu betrachtenden Aktiv-Zustand. Ist keine weitere koordinierte Transaktionsbearbeitung mehr möglich (z.B. wenn wesentliche Teile der Datenbasis ausgefallen sind), erfolgt ein Übergang in den einen Eingriff von außen erfordernden Ausfallzustand (5, 5s) (mit dem Ausfall einer Kontrollinstanz gelten auch die zugehörigen Konstitu-

enten als ausgefallen). Eine ausgefallene Kontrollinstanz geht ebenfalls in diesen Zustand über.

Existiert ein funktionsfähiges Restsystem, so ist eine weitere Transaktionsbearbeitung möglich, und es wird den anderen Kontrollinstanzen über W bzw. X angezeigt, in welchen Zustand von (4, rs) aus geordnet übergegangen werden soll.

Zustand (4, rs), r ε R, s ε S:
Der Übergang in den Zustand (1, rs) erfolgt, wenn die ausgefallene Kontrollinstanz keine Transaktionen in Bearbeitung hatte; ansonsten wird in den Zustand (1, 3, s) der Ebene für die Behandlung von Konstituentenausfällen (Rekonfiguration des Systems, Rücksetzung von Transaktionen usw.) übergegangen. Mit den Zustandsübergängen von (4, rs) in (1, rs) bzw. in (1, 3, s) wird der als vorläufig neu akzeptierte Aktiv-Zustand als endgültig angenommen und z_i entsprechend geändert. Bei Übergang in den Zustand (1, rs) wird auch der Aktiv-Zustand der Konstituenten entsprechend geändert.

Während eines Zyklus zur Behandlung von Kontrollinstanzausfällen können weitere Kontrollinstanzen ausfallen. Die Aktionen in diesem Zyklus sind deshalb auch Zeitüberwachungen zu unterwerfen. Der Übergang von (2, rs) in (3, rs) nach Erhalt von m_1 Bestätigungen für eine Fehlermeldung mit $m_1<m$ nach Ablauf eines time-out kann bewirken, daß in (3, rs) die Kontrollinstanzen unterschiedliche Ausfallmeldungen bearbeiten.

Mit den Nachrichten W bzw. X muß deshalb auch die nun bei der Kontrollinstanz vorliegende Information über den als vorläufig neu akzeptierten Aktiv-Zustand der Kontrollinstanzen mitgesendet werden. Nach Empfang von m Bestätigungen (m ergibt sich nun aus der Anzahl der vorläufig als aktiv akzeptierten Kontrollinstanzen) werden diese Zustandsinformationen miteinander verglichen. Falls die Kontrollinstanzen übereinstimmende Zustandsinformation besitzen und keine weiteren Fehlermeldungen zur Bearbeitung heranstehen, wird wie oben weiterverfahren.

Falls jedoch nur m_2 Bestätigungen mit $m_2<m$ (time-out) eingehen oder der Vergleich der Zustandsinformationen negativ verläuft, ist nach Übergang in den Zustand (1, rs) ein neuer Fehlerbehandlungszyklus zu starten, was durch die Nachricht f angezeigt ist.

Bei der Erstellung der neuen Fehlermeldung sind außerdem alle durch die wartenden Fehlernachrichten beantragten Zustandsänderungen mitzuberücksichtigen.

Bevor die Transaktionskoordinierung fortgesetzt oder neu begonnen werden kann, muß jeder Antrag auf Fehlerbehandlung behandelt worden sein. Das Verfahren endet entweder mit der erfolgreichen Bildung eines funktionsfähigen, die Weiterführung der Transaktionsbearbeitung sicherstellenden Teilsystems von Kontrollinstanzen oder mit dem Übergang aller Kontrollinstanzen in den Ausfallzustand (5, 5s).

Zur Einbettung in das TIL-System von 5.3. ist wie folgt vorzugehen. Der Zustand einer Kontrollinstanz i, i = 1(1)n, während eines Zyklus ist durch das Tupel

$$(y_i, z_i, q_i v_i, r_i u_i, s_i t_i)$$

beschrieben mit dem Vektor des Aktiv-Zustands der Kontrollinstanzen z_i, dem Vektor y_i für den als vorläufig neu akzeptierten Aktiv-Zustand bzgl. z_i, der Kombination aus Zustand der Protokolleinheit, repräsentiert durch q_i, und der zuletzt ausgesandten Nachricht v_i, die den Typ der Nachricht und die mit ihr übermittelte Zustandsinformation enthält, sowie mit $(r_i u_i, s_i t_i)$ wie in 5.3. Die Elemente von Σ entsprechen den möglichen Zuständen der Kontrollinstanzen, das Axiom w hat die Form

$$w = (\bar{1}, \bar{1}, a_1 0\bar{0}, a_1 0, a_1 0) \ldots (\bar{1}, \bar{1}, a_n 0\bar{0}, a_n 0, a_n 0)$$

wobei $\bar{0}$ den Nullvektor und $\bar{1}$ den Vektor (1,...,1) darstellen:
$P = \{P_1, P_2, P_3\}$ besteht aus drei Tabellen mit:

P_1 = Tabelle für die Koordinierung der Transaktionsbearbeitung,
P_2 = Tabelle für die Koordinierung des Ausfalls von Konstituenten,
P_3 = Tabelle für die Koordinierung des Ausfalls von Kontrollinstanzen, der aufgrund von Zeitüberwachungen "erkannt" wird.

P_1 und P_2 enthalten als Projektion die Produktionen von P_1 und P_2 von 5.2.; P_3 entsteht sinngemäß wie P_2 in 5.3.; das Skelett der Produktionen ist durch die Übergänge der Projektionen des Gesamtzustands bezüglich $(y_i, z_i, q_i v_i)$ definiert.

Zur Berücksichtigung der Ausgliederung von Kontrollinstanzen in dem <n-1, n-1>-TIL System G ist die Anwendbarkeit von Produktionen zur Bildung direkter Abbildungen einer Kontextrestriktion zu unterwerfen, die durch den Zustand der Interaktionsfähigkeit von Kontrollinstanzen bestimmt wird. Für eine Kontrollinstanz i, i = 1(1)n, ist die Menge der Kontrollinstanzen, mit der sie interagieren darf, durch die Menge $J_i = \{j: j \in \{1,...,n\}$ und $z_{ij} = 1\}$ gegeben. Die Produktionen in P müssen daher für alle möglichen Mengen J_i, i = 1(1)n, ausgelegt sein.

Die Korrektheit des Protokolls ergibt sich wegen der Separierung der Koordinierungsebenen unter den zu Beginn von 5.4. genannten Voraussetzungen aus

- der Korrektheit der Koordinierungsebenen für die Transaktionsbearbeitung und Behandlung des Ausfalls von Konstituenten,
- der Vorrangstellung der Behandlung von Ausfällen von Kontrollinstanzen gegenüber den anderen Koordinierungsebenen,
- der Korrektheit des Basisprotokolls, welche die Kooperation der Kontrollinstanzen im Zyklus für die Behandlung des Ausfalls von Kontrollinstanzen für den Fall sichert, daß keine Zeitschranken innerhalb dieses Zyklus überschritten werden,
- den Aktionen der Kontrollinstanzen, die erst dann eine Rückkehr in andere Koordinationsebenen erlauben, wenn ein funktionsfähiges Teilsystem mit identischen J_i existiert; kommt ein solches nicht zustande, so gehen alle Kontrollinstanzen in den Ausfallzustand über. Dies geschieht nach endlicher Zeit, da jedem Fehlerbehandlungsantrag einer als aktiv und damit als funktionsfähig geltenden Kontrollinstanz stattgegeben werden muß, d.h. mit jedem Zyklus verringert sich die Anzahl der als aktiv geltenden Kontrollinstanzen um mindestens eine Kontrollinstanz.
- Übergänge von einer Ebene zur anderen führen nur in erlaubte Konfigurationen des Protokollzustands, von denen eine weitere, korrekte Kooperation der Kontrollinstanzen des Restsystems aus gewährleistet ist.

Die Leistungsfähigkeit des vorgestellten Verfahrens ist wesentlich durch die Festlegung geeigneter Schranken für die time-outs bestimmt; sind sie zu klein gewählt, riskiert man die Bearbeitung ungerechtfertigter Ausfallmeldungen und damit das Auseinander-

streben der Kontrollinstanzen; sind sie zu groß vorgegeben, werden tatsächliche Ausfälle relativ spät erkannt und können einen erheblichen Mehraufwand bzgl. des Übergangs in ein funktionsfähiges Teilsystem implizieren.

5.4.2. Erweiterungen des Verfahrens

Das vorgestellte Verfahren ist zu modifizieren, falls die zu Beginn von 5.4.1. genannten Einschränkungen nicht gelten.

Der Ausfall von Einrichtungen des Nachrichtentransportsystems kann eine Zerlegung des Gesamtsystems in verschiedene isolierte, funktionsfähige Teilsysteme zur Folge haben. Die zu einem Teilsystem gehörenden Kontrollinstanzen müssen dann im Zustand (1, 3, s), in den sie unmittelbar nach Ausfallbehandlung übergehen, alle außerhalb ihres Teilsystems existierenden Konstituenten als nicht aktiv markieren und den weiteren verändernden Zugriff auf die in ihrem Teilsystem existierenden Objekte verbieten, sofern die Objekte auch in anderen Teilsystemen existieren können (redundante Realisierung).

Die nicht erkannte Verfälschung von Nachrichten durch das Nachrichtentransportsystem führt i.a. zum Ausbleiben von erwarteten Nachrichten bei anderen Kontrollinstanzen und damit letztlich zu einem time-out.

Arbeiten Kontrollinstanzen nicht korrekt, kann dies nur durch zusätzliche Maßnahmen zur Überprüfung von Aktivitäten in den Zuständen (3, rs), (1, 3, s) und (1, 1, 3) erkannt werden. Betrachten wir hierzu den Zyklus zur Behandlung von Ausfällen von Kontrollinstanzen. Initialisiert eine Kontrollinstanz die Durchführung eines solchen Zyklus, so ist die Initiierungsmeldung allen, auch den als ausgefallen "erkannten" Kontrollinstanzen zuzuleiten, um auch als vermeintlich ausgefallen betrachtete Kontrollinstanzen an der Abstimmungsphase zu beteiligen. Nach Eintritt in den Zustand (3, rs) wird der vorläufig akzeptierte neue Zustand y_i ermittelt, in dem als ausgefallen nur diejenige Kontrollinstanz betrachtet wird, die man selbst als ausgefallen erkannt hat (und für die man u.U. keine Eigeninitialisierung durchführen konnte). Im Zustand (4, rs) wird dann durch Vergleich aller eingehenden Zustandsmeldungen ein Zustandsvektor y_i' er-

mittelt, der die Mehrheitsentscheidung der Kontrollinstanzen repräsentiert. y_i und y_i' werden verglichen; bei Übereinstimmung erfolgt der entsprechende Übergang nach dem o.a. Protokoll, ansonsten muß ein neuer Zyklus beantragt werden. Ist eine Kontrollinstanz fehlerhaft, wird sie durch die anderen ausgegliedert; dies kann ihr aber auch widerfahren, falls die Mehrheit der Kontrollinstanzen fehlerhaft arbeitet. Im letzteren Fall muß eine Kontrollinstanz, falls sie ohne ersichtlichen Grund von den anderen Kontrollinstanzen ausgegliedert wurde, einen Eingriff von außen verlangen.
Da jedem Antrag auf Ausgliederung stattzugeben ist, kann das Verfahren zur unendlich-maligen Durchführung von Ausfallzyklen führen, falls eine fehlerhafte Kontrollinstanz sukzessiv ungerechtfertigte Ausfallbehandlungen beantragt; es ist deshalb zusätzlich eine Höchstzahl von unmittelbar aufeinander folgenden Zyklen vorzuschreiben, nach der ein Eingriff von außen zu verlangen ist.

5.5. Wiedereingliederung von Konstituenten und Kontrollinstanzen

Die Wiedereingliederung wiederhergestellter Konstituenten, deren Kontrollinstanz im funktionsfähigen Zustand verblieben und nicht ausgegliedert worden war, kann in einem alle nicht ausgegliederten Kontrollinstanzen eines Interaktionssystems umfassenden Koordinationszyklus in der Koordinierungsebene für die Transaktionsbearbeitung bewerkstelligt werden (eigenständige Koordinationsanforderung).

Die Wiedereingliederung einer reparierten Kontrollinstanz stellt für die übrigen Kontrollinstanzen eine Anforderung auf Koordinierung dar, deren Priorität niedriger als die Fehlerbehandlung, aber höher als die Transaktionsbearbeitung anzusehen ist. Wiedereingliederungsanforderungen unterschiedlicher Kontrollinstanzen müssen durch Prioritäten unterscheidbar sein. An der Wiedereingliederung sind alle Kontrollinstanzen des funktionsfähigen Interaktionssystems zu beteiligen, bei dem sie beantragt wird.

Die wiedereinzugliedernde Kontrollinstanz behandelt ihre Wiedereingliederung im Zyklus Q, der durch die Zustände {6, 7, 8} zu erweitern ist. Die Zustandübergänge für eine zu reintegrierende Kontrollinstanz sind in Bild 5.5 dargestellt.

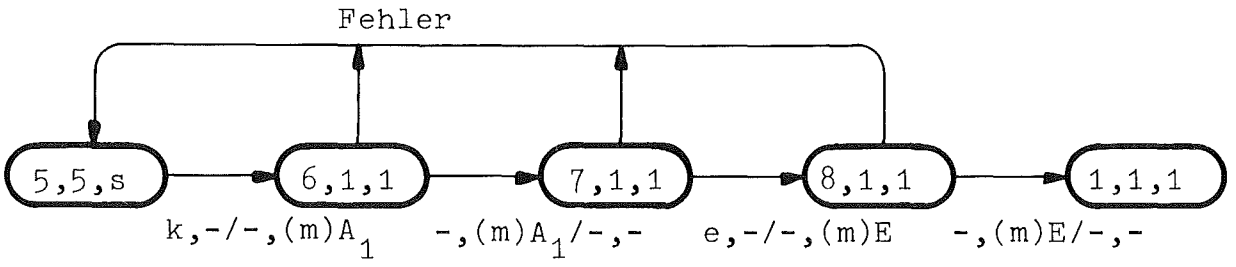


Bild 5.5: Protokollerweiterung der Kontrollinstanz 1 für ihre Wiedereingliederung

Sobald eine wiedereinzugliedernde Kontrollinstanz die rechnerinterne Präsenzmeldung k erhält, geht sie unter gleichzeitiger Übersendung der Wiedereingliederungsanforderung an die von ihr aus erreichbaren, zu demselben funktionsfähigen Interaktionssystem gehörenden, m aktiven Kontrollinstanzen in den Zustand (6, 1, 1) über. Die aktiven Kontrollinstanzen behandeln diese Anforderung in fester Kopplung in der Koordinierungsebene für die Transaktionsbearbeitung, d.h. sie führt zu Eigeninitialisierungen dieser Kontrollinstanzen. Es sei angenommen, daß das elementare Verfahren mit fester Kopplung (siehe 4.2) zur Transaktionsbearbeitung eingesetzt ist; die Zustandsmenge S ist dann um den Zustand {3''} zu erweitern. Sobald die aktiven Kontrollinstanzen der Wiedereingliederung in der Abstimmungsphase zustimmen, gehen sie in den Zustand (1, 1, 3'') bzw. die wiedereinzugliedernde Kontrollinstanz in den Zustand (7, 1, 1) über, in dem die Zustandsinformation sowie Konstituenten der Datenbasis aktualisiert werden. Danach kehren die aktiven Kontrollinstanzen über (1, 1, 4) bzw. die wiedereinzugliedernde über (8, 1, 1) in den Zustand (1, 1, 1) zurück, von dem aus die Transaktionsbearbeitung fortgeführt werden kann.

Tritt während des Reintegrationszyklus ein Fehler auf, so darf die wiedereinzugliedernde Kontrollinstanz nicht am Fehlerbehandlungszyklus teilnehmen, sondern geht in den Ausfallzustand zurück und beantragt zu einem späteren Zeitpunkt erneut ihre Wiedereingliederung.

Zur entsprechenden Erweiterung des formalen Modells von 5.4. ist u.a. eine neue Tabelle (P_4) hinzuzufügen, die die Produktionen für die Zustandsübergänge sowohl der aktiven Kontrollinstanzen als auch der wiedereinzugliedernden Kontrollinstanz umfaßt, da

die für einen Ableitungsschritt benötigten Produktionen alle aus derselben Tabelle zu entnehmen sind.

Die verklemmungsfreie Koordination der Wiedereingliederung durch die Kontrollinstanzen ist durch die Anwendung des vereinfachten Basisprotokolls bei den betroffenen Kontrollinstanzen gesichert, solange diese fehlerlos arbeiten.

Durch Maßnahmen der globalen Datensicherung ist zu gewährleisten, daß die Datenbasis bei Wiedereingliederung von Kontrollinstanzen und Konstituenten in einen konsistenten Zustand überführt werden kann.

Das Verfahren gestattet auch die Anwendung des Akkumulationsprinzips zur gleichzeitigen Wiedereingliederung mehrerer Kontrollinstanzen, sofern gesichert ist, daß die wiedereinzugliedernden Kontrollinstanzen miteinander kommunizieren können, da in diesem Fall alle Kontrollinstanzen für den Übergang in den Zustand (7,1,1) die gegenseitige Zustimmung benötigen.

Das Verfahren kann sinngemäß auch zur Wiederverschmelzung zweier funktionsfähiger, isolierter Interaktionssysteme angewendet werden, uzw. werden sie von den Kontrollinstanzen beider Interaktionssysteme in der Koordinationsebene für die Transaktionsbearbeitung abgehandelt, da vor der Wiederverschmelzung alle Kontrollinstanzen beider Interaktionssysteme aktiv und nicht ausgefallen waren.

6. Realisierung von Koordinierungsmechanismen

6.1. Integration von Kontrollinstanzen in Arbeitsrechnern

Die naheliegende Realisierung der entwickelten Koordinierungsmechanismen zur Transaktionsbearbeitung und Fehlerbehandlung besteht in der vollständigen Integration der Kontrollinstanzen in die Arbeitsrechner eines verteilten DV-Systems.

Eine wesentliche Grundlage hierfür ist der Grad an Komfort der Schnittstelle, die das Nachrichtentransportsystem den im verteilten DV-System existierenden (sequentiellen) Prozessen /B3,D4/ zur Interprozeßkommunikation mittels Nachrichtenaustausch bereitstellt.

Das Nachrichtentransportsystem sollte folgenden Forderungen genügen:

- die Kommunikationsprimitive zur Formulierung von Kommunikationsabläufen sollten sowohl für lokale als auch abgesetzte Kommunikation verwendbar sein,
- die Formulierung von Kommunikationsabläufen darf nicht von Implementierungsdetails abhängen, wie z.B. von der Struktur der Arbeitsrechner oder den speziellen Eigenschaften der physikalischen Einrichtungen zur Übertragung von Daten zwischen Arbeitsrechnern,
- der Namensraum zur Adressierung der Kommunikationspartner sollte weitgehend von dem Nachrichtentransportsystem verwaltet werden.

Von den existierenden Alternativen /A1/ von Nachrichtentransportsystemen eignet sich für die Kooperation von Kontrollinstanzen insbesondere das Konzept der verbindungsorientierten Kommunikation.

Bei diesem Konzept wird vor der Übertragung von Nachrichten zwischen zwei Kommunikationspartnern eine gerichtete logische Verbindung aufgebaut. Über diese Verbindung können beliebig viele Nachrichten in der entsprechenden Richtung übertragen werden; der Abbau der Verbindung erfolgt erst, wenn die Verbindung nicht mehr benötigt wird.

In jedem Arbeitsrechner existiert (mindestens) eine Transportstation, die eine Menge von logischen Anschlußstellen (Ports /C1/)

verwaltet. Die Gesamtheit der Ports bildet den Namensraum für die Interprozeßkommunikation. Jeder der Transportstation zugeordneten Prozesse darf über die Transportstation Anschlußstellen als logische Eingänge (Empfangs-Ports) für den Empfang von Nachrichten bzw. logische Ausgänge (Sende-Ports) für das Aussenden von Nachrichten benutzen. Sobald eine Zuordnung von Sende-Port s eines Prozesses P_1 und Empfangs-Port e eines Prozesses P_m - geordnetes Paar (s,e) - existiert, ist die gerichtete logische Verbindung von P_1 zu P_m hergestellt (siehe Bild 6.1).

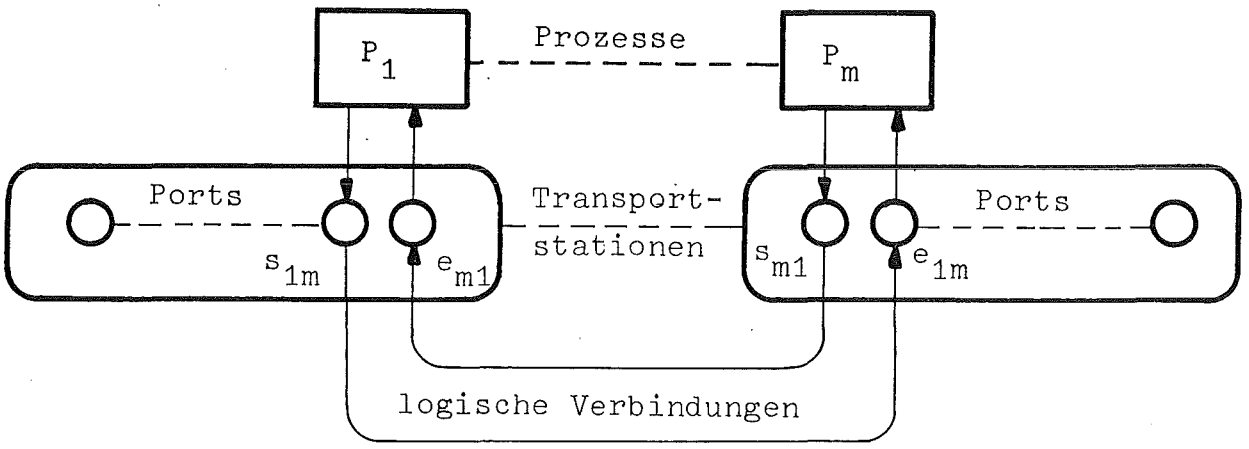


Bild 6.1: Logische Verbindungen von Prozessen über Ports unter der Verwaltung von Transportstationen

Den Prozessen stehen im wesentlichen folgende, in vereinfachter Form aufgeführte Kommunikationsprimitive zur Verfügung:

- ERÖFFNE VERBINDUNG (Art, Port, Ereignisvariable)
  Durch <Art> wird die aktive bzw. passive Eröffnung einer Verbindung angezeigt. Bei aktiver Eröffnung sind die Parameter der eigene Sende-Port und der fremde Empfangs-Port; bei passiver Eröffnung sind der eigene Empfangs-Port und eine Ereignisvariable z.B. zur Anzeige einer Nachrichtenankunft als Parameter zu führen. Eine gerichtete logische Verbindung von P_1 zu P_2 kommt nur zustande, wenn P_1 aktiv und P_2 passiv die Verbindung eröffnen,
- SENDE (eigener Sende-Port, Nachricht) zum Senden einer im Nachrichtenpuffer des Prozesses stehenden Nachricht über eine bestehende logische Verbindung,

- EMPFANGE (eigener Empfangs-Port, Nachricht) zur Übernahme einer Nachricht über eine bestehende logische Verbindung in den Nachrichtenpuffer des Prozesses,
- BEENDE VERBINDUNG (eigener Port)
  zum Abbau einer nicht mehr benötigten Verbindung.

Die Aufgabe der Transportstationen ist die Verwaltung der Zuordnung der logischen Verbindungen zu den Prozessen und die Bereitstellung von Dienstleistungen für die Übertragung von Nachrichten auf den logischen Verbindungen, z.B. Numerierung von Nachrichten, Fehlerkontrolle (Prüfsummenbildung), Flußkontrolle, Fragmentierung von Nachrichten in Teilnachrichten usw. /C1/. Die korrekte Abwicklung dieser in Kooperation der Transportstationen durchzuführenden Aufgaben ist Gegenstand des Transportprotokolls.

Zur Realisierung der Interprozeßkommunikation bedarf es i.a. einer Hierarchie von Kommunikationsprotokollen. In dieser werden, in Analogie zu schichtenförmig organisierten Betriebssystemen, die für eine Kommunikationsebene bereitgestellten elementaren Kommunikationsfunktionen durch Protokolle der darunterliegenden Schicht implementiert. Die Protokollhierarchie in Nachrichtentransportsystemen läßt sich im einfachsten Fall in zwei Schichten /P1/ organisieren (siehe Bild 6.2), indem dem Transportprotokoll (Schicht S2) eine weitere Schicht S1 unterlegt wird.

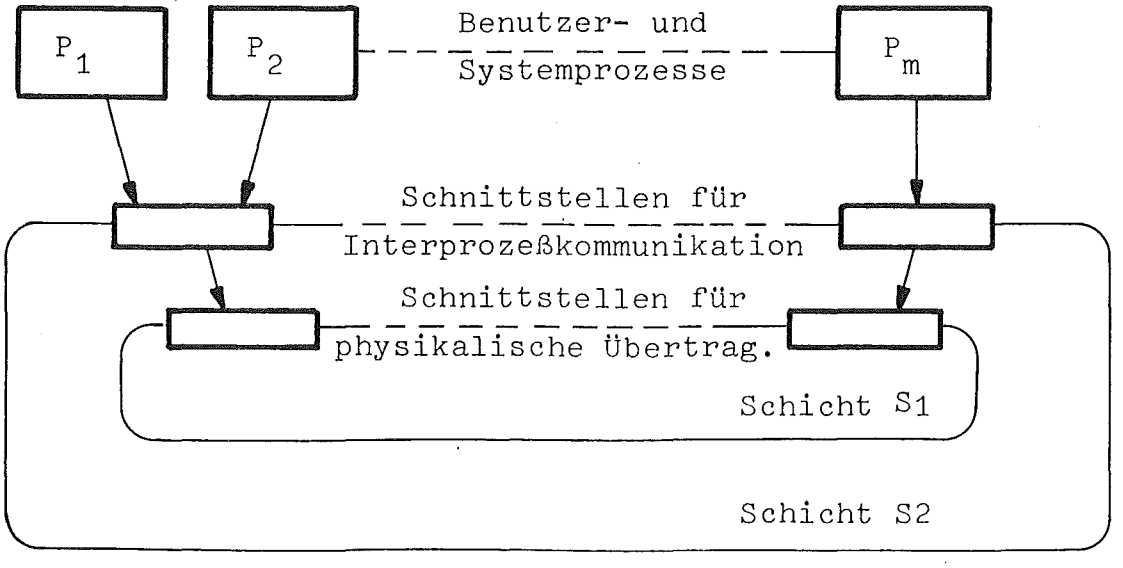


Bild 6.2: Schichtenstruktur eines Nachrichtentransportsystems

Die Schicht S1 liegt der Hardware am nächsten und unterstützt den Datenaustausch zwischen unmittelbar miteinander verbundenen Rechnern (data link control procedure), d.h. falls Arbeitsrechner unmittelbar miteinander gekoppelt sind, regelt sie den Austausch zwischen diesen Arbeitsrechnern; sind Arbeitsrechner über Kommunikationsrechner miteinander verbunden, so steuert sie den Datenaustausch Arbeitsrechner-Kommunikationsrechner sowie zwischen Kommunikationsrechnern (siehe Bild 6.3). Ihr obliegt die Verwaltung der physikalischen Kommunikationswege.

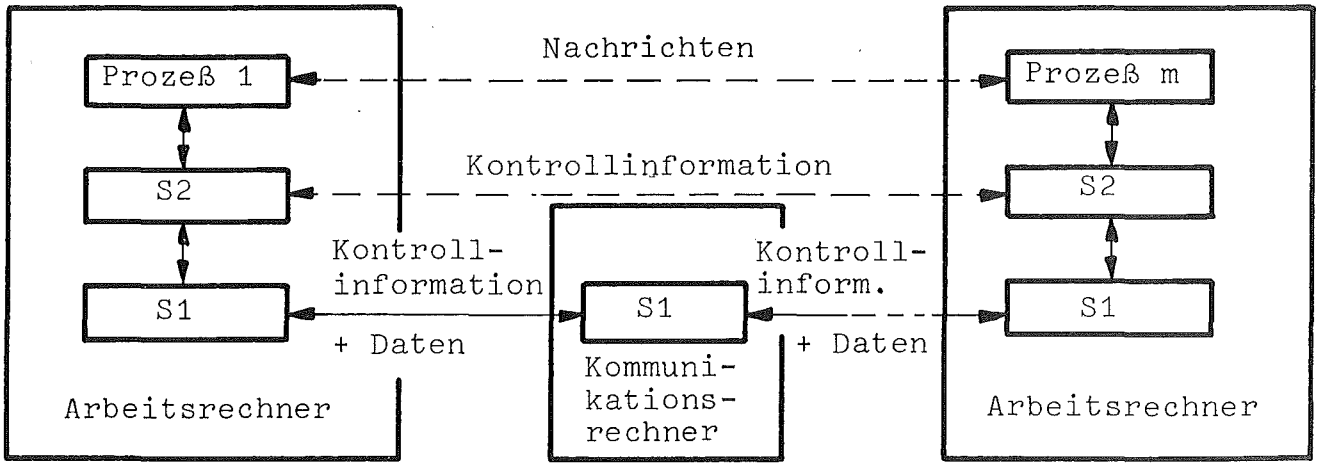


Bild 6.3: Informationsfluß in Rechnernetzen
(⟷ real, ⟷ virtuell)

Die Protokolleinheiten unterschiedlicher Schichten in einem Rechner kommunizieren mittels Kommandos bzw. Rückmeldungen; in diesen Richtungen findet auch der reale Informationsfluß statt (Bild 6.3). Protokolleinheiten derselben Protokollschicht auf verschiedenen Rechnern kreieren bzw. interpretieren nur für diese Ebene verbindliche, formatierte und durch das Protokoll in ihren Bedeutungen festgelegte Kontrollinformationen. Angekommene Daten und Kontrollinformationen, die für eine höhere Schicht bestimmt sind, werden immer vollständig weitergeleitet und nicht interpretiert. Die maximale Länge einer Übertragungseinheit für Daten ist in S2 meistens systemweit konstant, in S1 können zwischen unterschiedlichen Paaren von Protokolleinheiten verschiedene maximale Längen existieren (z.B. in Abhängigkeit von der Kapazität der Rechnerkopplung).

Zur Implementierung einer Kontrollinstanz unter Verwendung der durch das Nachrichtentransportsystem bereitgestellten Funktionen für die Interprozeßkommunikation sind zusätzliche Funktionen zu berücksichtigen:

- WARTE (Ereignisvariable): diese Funktion dient der internen Synchronisation einer Kontrollinstanz, um die Abhängigkeit der weiteren Vorgehensweise von Ereignissen wie Nachrichtenankunft oder Time-out ausdrücken zu können.
- SETZE_TIMEOUT (Identifikation, Intervall), LÖSCHE_TIMEOUT (Identifikation) für das Aktivieren bzw. Außerkraftsetzen von Zeitschranken für Aktionen.

Die grobe Ablaufstruktur des Prozesses "Kontrollinstanz" ist wie folgt:
Zunächst durchläuft die Kontrollinstanz eine Initialisierungsphase, innerhalb der sie die deklarierten Variablen (Zustandsvariablen, Ereignisvariablen usw.) vorbesetzt und die aktive und passive Eröffnung der logischen Verbindungen zu ihren Kommunikationspartnern (Kontrollinstanzen, Benutzerprozesse usw.) herstellt. Danach wartet sie auf das Eingehen von Nachrichten (WARTE (nachrichtenankunft)), die sie einzeln mittels Aufruf der Funktion EMPFANGE in ihre Nachrichtenwarteschlange überführt. Sie überprüft, ob die Nachricht eine Aktion z.B. der Protokolleinheit erfordert und veranlaßt die Aktionsausführung. Nach der Aktion kehrt die Kontrollinstanz in den Wartezustand zurück.

Im folgenden wird die Realisierung des vereinfachten Basisprotokolls für den S-Zyklus (Transaktionsbearbeitung) für Kontrollinstanz 1 schematisch aufgezeigt.

```
*Deklaration der Variablen*
   type status = (s1,s2,s3,s4);
        eingabenachricht = (a,v,A,e,E);
        wertebereich = (s1_a,s1_v,s1_A,s2_A,s3_e,s4_E,undef);
   var i: status; j: eingabenachricht;
       n: array (s1..s4, a..E) of wertebereich;
--------
*Vorbesetzung von Variablen*
   n(s1,a):= s1_a;.....; n(s4,E):= s4_E;
   *irrelevante Werte von n:= undef*
-------
```

```
*S-Zyklus (Transaktionsbearbeitung)*
   case n(i,j) of
      s1_a: *Eigeninitialisierung*
            for k:= 2(1)n do SENDE (s-port(1,k), A1);
            SETZE_TIMEOUT (1, intervall);
            i:= s2;
      s1_v: *Eigeninitialisierung*
            --- entsprechend s1_a ---
      s1_A: *Fremdinitialisierung*
            --- entsprechend s1_a ---
      s2_A: if <Verdrängung erlaubt> then
               begin LÖSCHE_TIMEOUT(1);
                     for k:= 2(1)n do SENDE (s-port(1,k),Aip);
                     SETZE_TIMEOUT (2,intervall);
               end;
            if <n-1 Ai* eingetroffen> then
               begin LÖSCHE_TIMEOUT (2);
                     i:= s3;
                     <veranlasse die Bearbeitung der Koordinations-
                      anforderung>;
               end;
      s3_e: *Koordinationsanforderung bearbeitet*
            for k:= 2(1)n do SENDE (s-port (1,k), E);
            SETZE_TIMEOUT (3,intervall);
            i:= s4;
      s4_E: *alle n-1 Endemeldungen eingetroffen*
            LÖSCHE_TIMEOUT (3);
            <Endebehandlung zur Vorbereitung eines neuen
             S-Zyklus veranlassen>;
            i:= s1;
      undef: *Fehler*
   end;
*Ende von S-Zyklus*
```

6.2. Realisierung von Koordinationsfunktionen mit spezieller Hardware

Die vollständige Integration von Kontrollinstanzen in die Arbeitsrechner bedeutet gerade im Falle der Verfahren mit fester Kopplung eine nicht unbeträchtliche Belastung der Arbeitsrechner. Neben der Abwicklung des Protokolls, z.B. Durchführen von Verdrängungen in der Abstimmungsphase, müssen zusätzlich Time-out-Schranken zur Überwachung protokollbezogener Aktivitäten berücksichtigt werden. Die Festlegung optimaler Time-out-Schranken hängt von sehr vielen Faktoren ab, z.B. von

- der prioritätsmäßigen Einbettung der Kontrollinstanz im Arbeitsrechner,
- der inhärenten Übertragungsverzögerung durch das Nachrichtentransportsystem, die sich im wesentlichen ergibt aus der Übertragungsgeschwindigkeit und der Zuverlässigkeit der physikali-

schen Übertragungswege sowie der Effizienz der einzelnen Schichten der Protokollhierarchie.

Sie kann daher nur sehr grob erfolgen; werden die Schranken zu groß gewählt, wird die Reaktion auf Fehler verzögert; wählt man sie zu klein, sind zeitaufwendige Bearbeitungen ungerechtfertigter Ausfallzyklen die Folge.

Es empfiehlt sich, gerade für Realzeitanwendungen, Teile des Koordinierungsmechanismus in speziellen Komponenten des Nachrichtentransportsystems zu realisieren; insbesondere bietet sich eine Integration von Koordinationsfunktionen in Kommunikationsprozessoren an.

Dies kann in der Form spezieller Koordinations-Tasks, Sekretäre /D5/, geschehen, die eine ähnliche Struktur wie die Kontrollinstanzen aufweisen und für die Interkommunikation äquivalente Protokolle benutzen (siehe Bild 6.4).

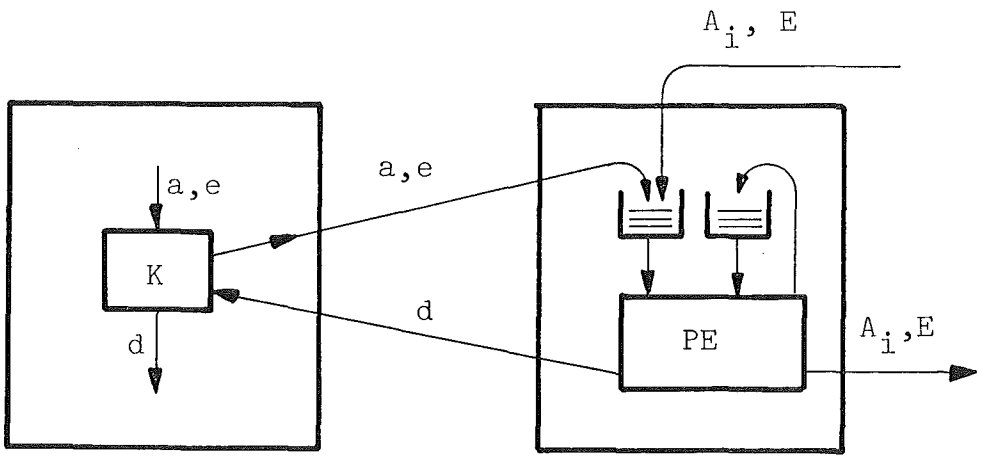


Bild 6.4: Zusammenfassung von Koordinationsfunktionen in einem Sekretär im Kommunikationsrechner (K = Kontrollinstanz, PE = Protokolleinheit)

Unter Zugrundelegung des vereinfachten Basisprotokolls von 3.2. bietet sich folgende Vorgehensweise an:

Eine Kontrollinstanz versieht eine zu bearbeitende Koordinationsanforderung mit einer systemweit eindeutigen Priorität und übergibt diese ihrem Sekretär. Dieser führt in Kooperation mit den anderen Sekretären die Abstimmungsphase durch und unterrichtet seine Kontrollinstanz über die nach Abschluß der Abstimmungspha-

se auszuführende Koordinationsanforderung. Nach Beendigung des kritischen Abschnitts durch die Kontrollinstanz wickelt ihr Sekretär die Restphase ab, deren Ende er der Kontrollinstanz u.U. nicht notwendig zurückmelden muß /H7/.

Dieses Verfahren ist jedoch mit einem schwerwiegenden Nachteil behaftet. Um bei nicht leerer Warteschlange für Koordinationsanforderungen unabhängig von der Kontrollinstanz einen neuen Zyklus initiieren zu können, ist bei einer Anforderung auf Betriebsmittelbelegung zu überprüfen, ob die benötigten Betriebsmittel verfügbar sind. Diese Information wird aber noch bei der Kontrollinstanz im Arbeitsrechner geführt.

Abhilfe schafft hier nur eine weitere Abmagerung der Kontrollinstanz im Arbeitsrechner, in dem die gesamte Kontrolle und Manipulation der globalen Zustandsinformation dem Sekretär übertragen wird. Dies bedeutet einerseits, daß bei dieser Alternative ein Sperrmechanismus der Art - physische Sperrung mit Spezifikation der Betriebsmittel über Prädikat - nicht sinnvoll angewendet werden kann (die anderen Sperrmechanismen lassen sich gut einsetzen), andererseits, daß lokal bearbeitbare Transaktionen (deren Bearbeitung nicht durch mehrere Kontrollinstanzen koordiniert werden muß) sich bei dem Sekretär um die Betriebsmittelzuweisung bewerben müssen. Letzteres beinhaltet eine unzulässige Diskriminierung lokal bearbeitbarer Transaktionen, falls nicht für eine Bevorzugung ihrer Bearbeitung unter gleichzeitiger Anwendung des Verfahrens mit loser Kopplung und Verklemmungsbeseitigung (siehe 4.3.2) gesorgt wird.

Da zudem ein Kommunikationsrechner die Aufgabe hat, den Status der Verbindungen zu allen bei ihm angeschlossenen Stationen zu überwachen, kann die Zuständigkeit für die Time-out-Kontrolle der protokollgesteuerten Aktionen ebenfalls dem Sekretär übertragen werden. Situationen wie die Auftrennung des verteilten DV-Systems in Teilsysteme oder die Nichterreichbarkeit von Arbeitsrechnern bei Ausfall von Rechnern oder Kommunikationseinrichtungen werden schneller erkannt. Diese Lösung erlaubt die Festlegung kleinerer und besser angepaßter Zeitschranken als in 6.1. Verwendbare Abschätzungen für Laufzeiten von Nachrichten zwischen Kommunikationsrechnern bei Anwendung des store-and-forward-Verfahrens zur Nachrichtenübermittlung sind in /M3,S4/ zu finden.

6.3. Untersuchung der Leistungsfähigkeit der Realisierungsvarianten

Eine Untersuchung der Leistungsfähigkeit der beiden Realisierungsvarianten für Kontrollinstanzen ist insbesondere für Protokolle mit fester Kopplung interessant, da dieses Prinzip der Vorgehensweise zur koordinierten Reaktion auf Störsituationen zugrunde liegt. Die Beschränkung auf das vereinfachte Basisprotokoll von 3.2. ist dabei für die Untersuchung ausreichend; die eigentliche Ausführung der mit einer Koordinationsanforderung assoziierten Aktivität im Zustand 3 des Basisprotokolls ist für den Vergleich der Varianten irrelevant.

Ziele der Untersuchung der Leistungsfähigkeit der Varianten sind qualitative und quantitative Aussagen über Unterschiede in folgenden Leistungskenngrößen:

- Dauer von Koordinationszyklen,
- Auslastung der Arbeitsrechner und Kommunikationsrechner durch die Bearbeitung von Koordinationsanforderungen und der durch Koordinationszyklen implizierten Kontrollinformationen,
- Intensität der ungerechtfertigten Störungen der Transaktionsbearbeitung bei Vorgabe unterschiedlicher Zeitschranken (timeouts) zur Einleitung von Fehlerbehandlungszyklen,

in Abhängigkeit von der Belastung des Nachrichtentransportsystems durch Ströme von Teilnachrichten (packets), die nicht mit der Tätigkeit der Kontrollinstanzen zusammenhängen.

Verteilte DV-Systeme als auftragsbearbeitende Systeme lassen sich vorteilhaft als Warteschlangensysteme modellieren /H2,H3/, um entweder in Spezialfällen analytische Ergebnisse der Warteschlangentheorie zur Aussage über die Leistungsfähigkeit der verteilten DV-Systeme unmittelbar anwenden oder um über eine Implementierung des Systems als Simulationsmodell in einer Simulationssprache (z.B. SIMULA /D1/) entsprechende Ergebnisse auf experimentellem Weg gewinnen zu können.

Die Wartezeit von Koordinationsanforderungen bei fester Kopplung der Kontrollinstanzen läßt sich durch analytische Ergebnisse der Warteschlangentheorie approximieren /K4/, falls Mittelwert und Varianz der Dauer von Koordinationszyklen bekannt sind. Man geht

von folgendem Modell aus: Eine Bedienungsstation repräsentiert das verteilte DV-System und n Warteschlangen repräsentieren die n Prioritätsklassen von Koordinationsanforderungen; bei poissonverteilten Ankünften der Koordinationsanforderungen können die Wartezeiten für die Prioritätsklassen gesondert bestimmt werden.

Zur Entwicklung eines Warteschlangenmodells zur Untersuchung der o.a. Leistungskenngrößen muß die Modellierungstiefe bestimmt werden. Bei der Forderung nach Quantifizierbarkeit des Unterschieds der Realisierungsvarianten ist eine Berücksichtigung der detaillierten Arbeitsweise der Protokollschichten des Nachrichtentransportsystems unabdingbar. Eine Protokollschicht in einem Rechner besteht dabei üblicherweise aus einem Modul für die Eingabe- und einem Modul für die Ausgabe-Richtung; beide Moduln befinden sich in starken, protokollbedingten Wechselbeziehungen. Mit dem Warteschlangenmodell einer Kontrollinstanz und der Einteilung der Protokollhierarchie des Nachrichtentransportsystems in zwei Schichten erhalten wir das Warteschlangenmodell von Bild 6.5 für die Instanzen in einem Arbeitsrechner für die Variante der vollständigen Integration einer Kontrollinstanz im Arbeitsrechner; es gilt auch für einen Kommunikationsrechner bei der Sekretärstechnik-Variante.

Kontrollinstanz sowie Protokollmodule der Schichten S1 und S2 bewerben sich um das Betriebsmittel Rechner, d.h. sie sind selbst wieder als Elemente eines (geschlossenen) Warteschlangensystems aufzufassen. Rechnerkopplungen können als Warteschlangensystem mit einer Bedienungsstation und rechnerseitig assoziierten Warteschlangen modelliert werden.

Wegen der starken impliziten und expliziten Wechselwirkungen der Funktionseinheiten der Rechner ist eine Implementierung des Gesamtmodells als Simulationsmodell erforderlich, da analytische Auswertungen unzulässige Restriktionen bzgl. der Wechselwirkungen von Protokollschichten voraussetzen. Um validierbare Aussagen zu erhalten, müssen die Aktionen der Protokollschichten mit denen eines realen verteilten DV-Systems übereinstimmen.

Die o.a. Modellierung der Protokollschichten wurde bei der Modellierung (in SIMULA) des Nachrichtentransportsystems eines existierenden Rechnernetzes /S5/ angewendet; die Funktionen der Mo-

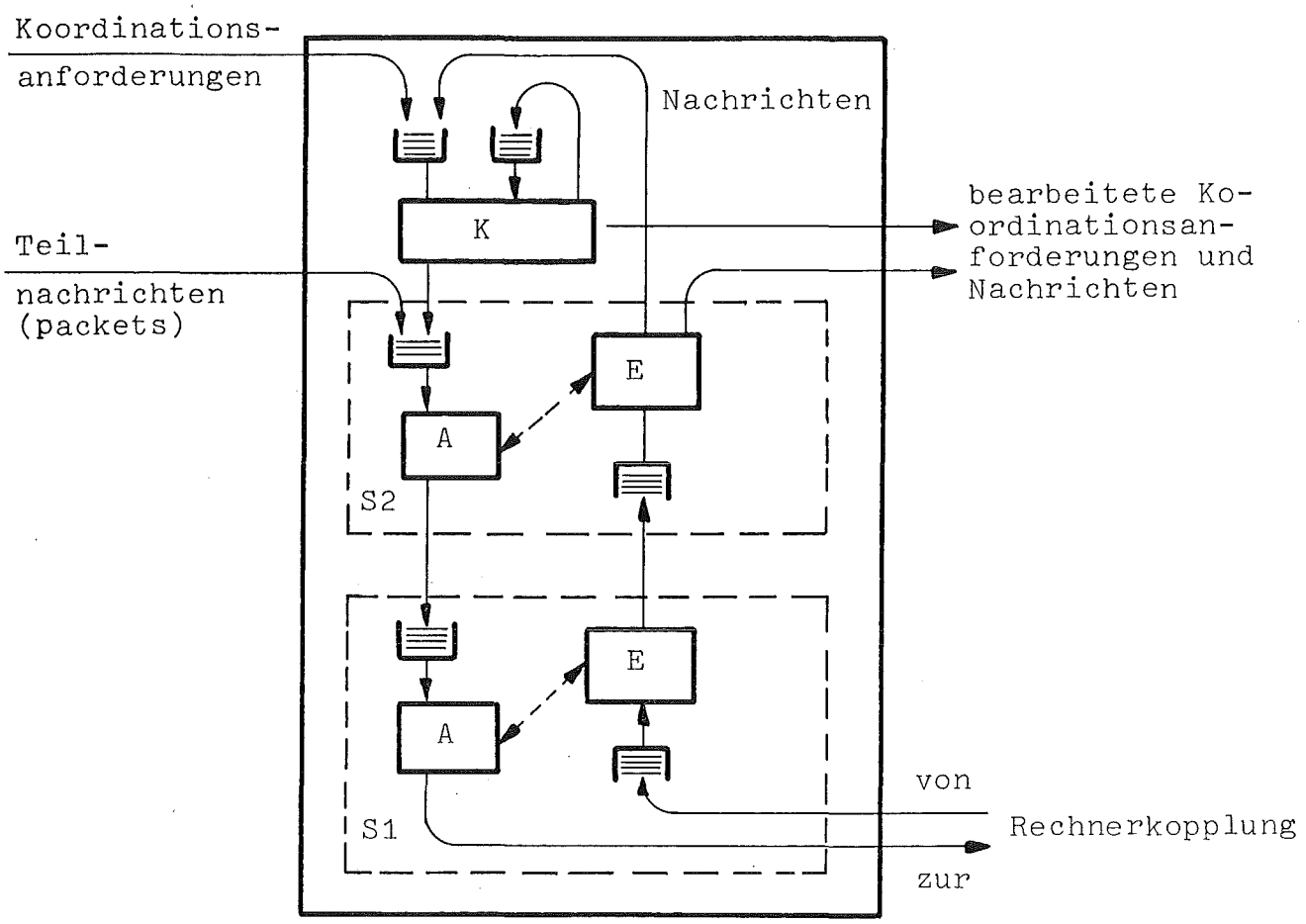


Bild 6.5: Warteschlangensystem der Funktionseinheiten eines Rechners (K = Kontrollinstanz, A = Ausgabemodul, E = Eingabemodul)

dule wurden im Detail berücksichtigt. In dieses realitätsnahe, validierte Modell wurden Modelle der beiden Realisierungsvarianten integriert und einer experimentellen Effizienzuntersuchung unterzogen.

Die Experimente basieren auf einer Konfiguration von 4 Prozeßrechnern, die über 170 [K Bytes/sec] schnelle Kanal-Kanal-Kopplungen mit ihren assoziierten Kommunikationsrechnern gekoppelt sind (Bild 6.6).

Die 4 Kommunikationsrechner sind miteinander vollständig über Leitungen mit einer um den Faktor 10 kleineren Übertragungskapazität verbunden. Bei den zu simulierenden Softwareaktivitäten sind die an dem o.a. existierenden Rechnernetz /S5/ ermittelten Werte für Ausführungszeiten verwendet. In Anlehnung an diese sind auch die Zeiten für die Aktionen der Kontrollinstanzen und Sekretäre ge-

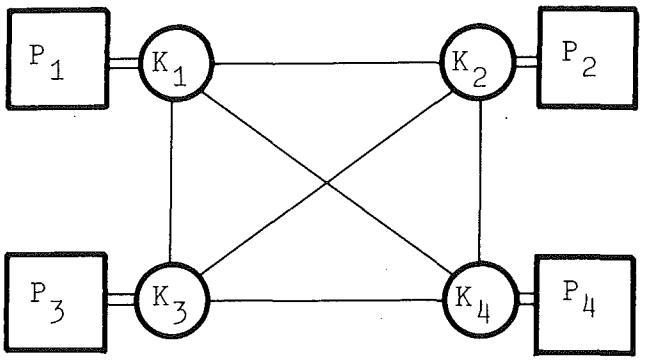


Bild 6.6: Konfiguration des simulierten Rechnernetzes
(P = Prozeßrechner, K = Kommunikationsrechner)

wählt. Die Länge der Teilnachrichten entstammen einer Gleichverteilung in den Grenzen 100-500 [Bytes], die Länge der Kontrollnachrichten des Basisprotokolls ist konstant mit 10 [Bytes] angenommen. Andere Aktivitäten als die des Nachrichtentransportsystems und der Kontrollinstanzen wurden nicht berücksichtigt, um Verzerrungen der Ergebnisse auszuschließen.

Im Rahmen von Pilotläufen wurde die Grenze für die Stationarität der beiden Realisierungsvarianten sowie die Einschwingphase ermittelt. Für die Stichproben, auf deren Basis die Mittelwerte, die 95%-Konfidenzintervalle sowie die 95%-Signifikanz für den Unterschied von Mittelwerten bestimmt wurden, konnten Unabhängigkeit, Stationarität und Normalverteilung nachgewiesen werden /D6/.

Die Ergebnisse - Bild 6.7 - 6.10 - zeigen eine deutliche Überlegenheit der Sekretärstechnik-Variante gegenüber der Arbeitsrechner-Variante, die im Fall a = 80 [Teilnachrichten/sec] generell instationäres Verhalten zeigte.

Die Belastung - siehe Bild 6.7 - eines Prozeßrechners bzw. Kommunikationsrechners durch Koordinationsanforderungen betrug 0.04. Dies zeigt, daß die hohe Auslastung des Prozeßrechners (Bild 6.7a) bei der Arbeitsrechner-Variante aus der Belastung des Nachrichtentransportsystems durch den Koordinationsnachrichtenfluß resultiert. Dieser Effekt ist auch bei der Auslastung des Kommunikationsrechners (Bild 6.7b) zu beobachten. Für die hohen Aus-

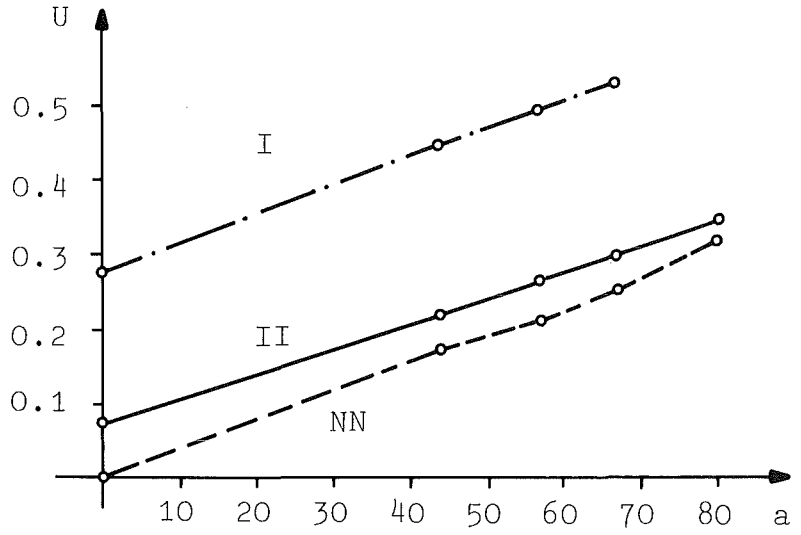
lastungswerte ist in beiden Fällen die Schicht S1 verantwortlich. Die gegenüber der Arbeitsrechner-Variante geringere Auslastung des Kommunikationsrechners durch die Sekretärstechnik-Variante ergibt sich aus der reduzierten Belastung der Schicht S1 zwischen Kommunikationsrechner und Prozeßrechner.

Bild 6.8 gibt eine Übersicht über die mittlere Dauer eines Koordinationszyklus für beide Varianten bei unterschiedlichen Belastungen des Nachrichtentransportsystems.

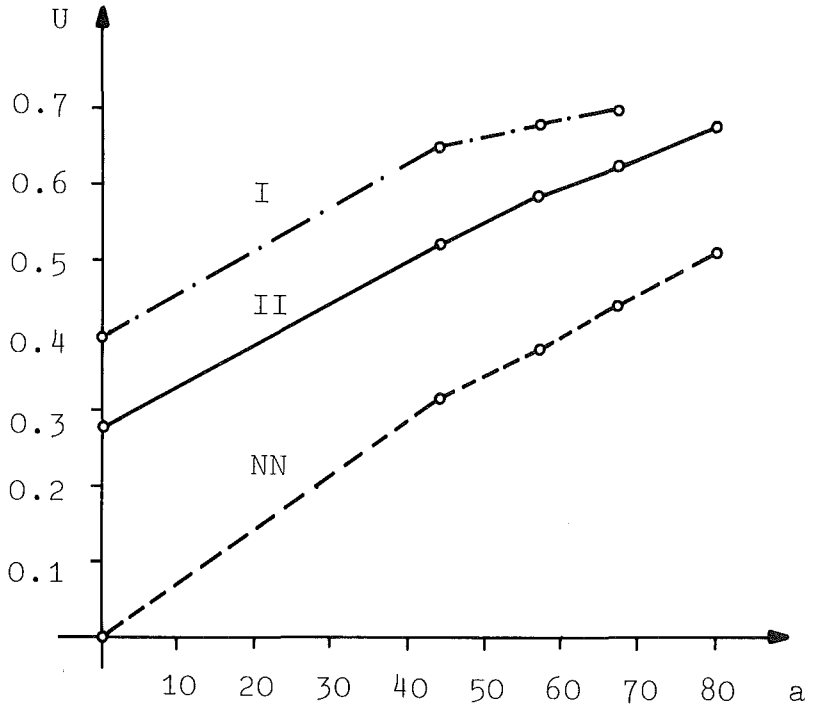
Bild 6.9 vermittelt einen Einblick in die mittlere Verweilzeit von Nachrichten im Nachrichtentransportsystem; diese Ergebnisse können zur rohen Abschätzung der Time-out-Schranken für die Überwachung der Protokollaktivitäten herangezogen werden (vgl. Bild 6.10).

Bild 6.10 zeigt, daß die Zeitschranken für die Arbeitsrechner-Variante zu groß angesetzt sind; T = 4 bedeutet nämlich, daß im Mittel in jedem Koordinationszyklus eine Initialisierung eines ungerechtfertigten Fehlerbehandlungszyklus erfolgen würde.

Die günstigen Eigenschaften der Sekretärstechnik-Variante spricht insbesondere für ihren Einsatz in Realzeitsystemen, um eine effiziente Führung verteilt realisierter Prozeßdatenbasen zu unterstützen.



(a) Prozeßrechner



(b) Kommunikationsrechner

Bild 6.7: Abhängigkeit der Auslastung U von Prozeßrechner (a) und Kommunikationsrechner (b) von der Ankunftsrate a von Teilnachrichten/sec im Nachrichtentransportsystem und der Ankunftsrate b=4 Koordinationsanforderungen/sec für die Varianten I-Integration in Arbeitsrechner und II-Sekretärstechnik. NN stellt das System ohne Koordinationsanforderungen dar.

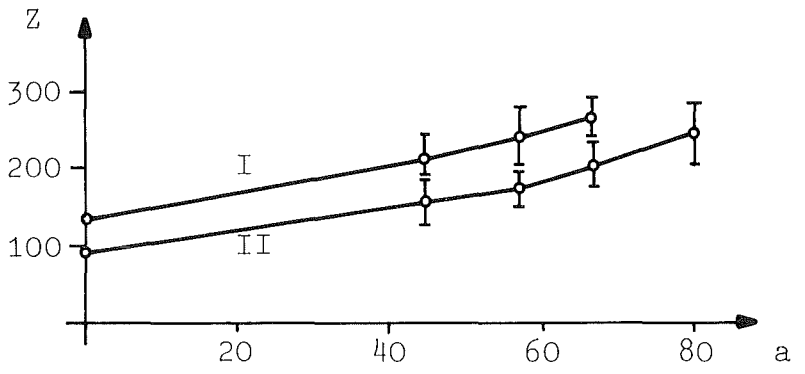


Bild 6.8: Abhängigkeit der mittleren Zykluszeit Z [msec] von der Ankunftsrate a von Teilnachrichten/sec im Nachrichtentransportsystem und der Ankunftsrate b=4 Koordinationsanforderungen/sec für die Varianten I - Integration in Arbeitsrechner und II - Sekretärstechnik.

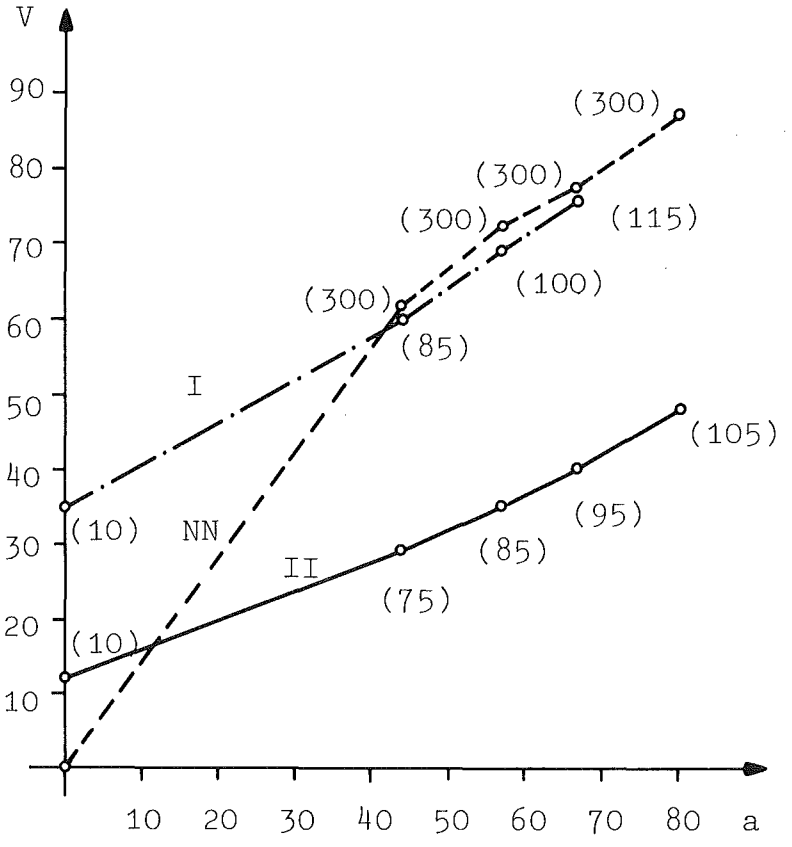


Bild 6.9: Abhängigkeit der mittleren Verweilzeit V [msec] von Teilnachrichten im Nachrichtentransportsystem von der Ankunftsrate a von Teilnachrichten/sec und der Ankunftsrate b=4 Koordinationsanforderungen/sec für die Varianten I-Integration in Arbeitsrechner und II-Sekretärstechnik. NN stellt das System ohne Koordinationsanforderungen dar. In Klammern sind die mittleren Nachrichtenlängen beigefügt.

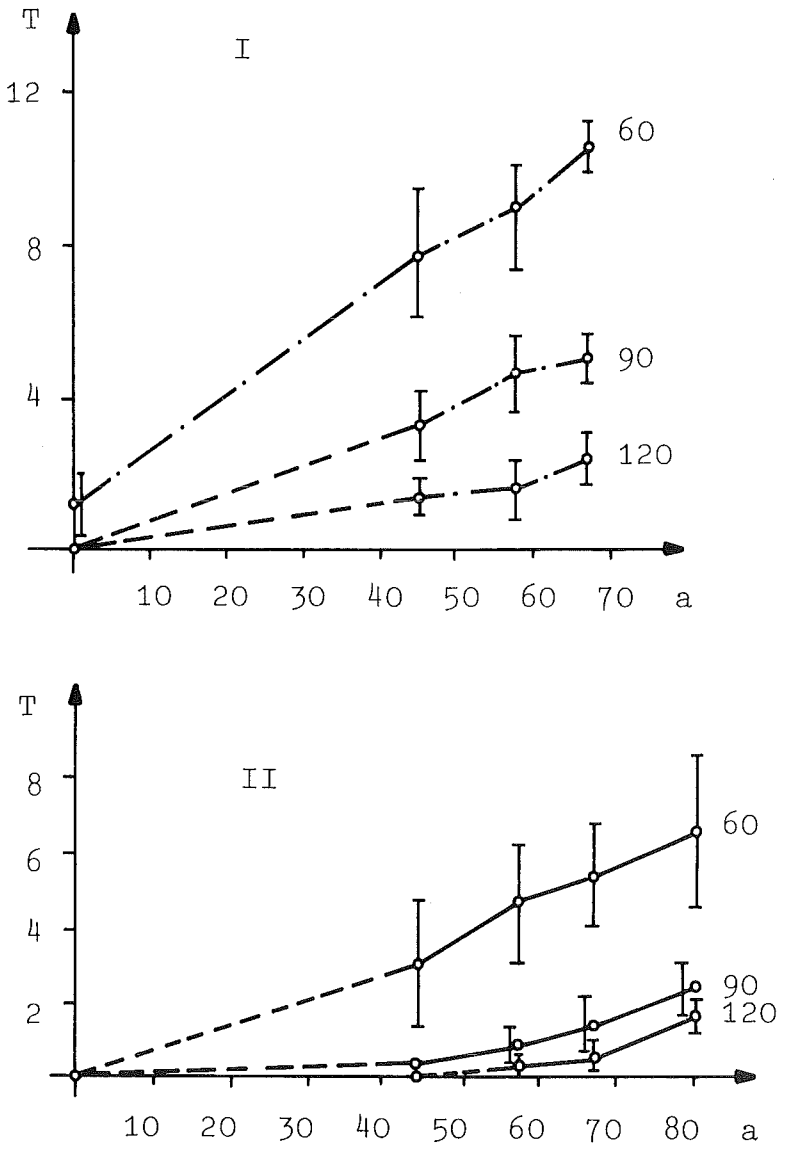


Bild 6.10: Abhängigkeit der mittleren Anzahl T ungerechtfertigter Initialisierungsversuche für Fehlerzyklen aufgrund von Zeitschrankenüberschreitungen von der Ankunftsrate a von Teilnachrichten/sec im Nachrichtentransportsystem, der Ankunftsrate b=4 Koordinationsanforderungen/sec und den Zeitschranken 60 [msec] 90 [msec] und 120 [msec] für die Varianten I-Integration in Arbeitsrechner und II-Sekretärstechnik.

7. Zusammenfassung und Ausblick

Der zunehmende Bedarf an integrierter Bearbeitung komplexer Aufgabenbereiche, z.B. in der Lenkung technischer Prozesse, durch verteilte DV-Systeme bedingt aus Effizienz- und Zuverlässigkeitsgründen eine Realisierung logisch zusammengehörender Informationsbestände in Form verteilter Datenbanken mit einer dezentralen Kontrollstruktur.

Ein wichtiger Gesichtspunkt ist die Sicherung der Integrität der in der Datenbank abgespeicherten Information durch Zugriffskontrolle von operationaler Seite aus, um gegenseitige Störungen und Konsistenzverletzungen durch parallel die Datenbasis manipulierende Benutzer zu verhindern.

Für zentralisierte Datenbanken entwickelte Sperrmechanismen sind in verteilten DV-Systemen mit zentraler Kontrollstruktur einsetzbar, bei der jeder Zugriffswunsch durch die zentrale Kontrollinstanz bearbeitet wird. Für verteilte Datenbasen mit dezentralisierter Kontrolle durch ein System funktionell äquivalenter Kontrollinstanzen existieren erst Lösungsansätze für den Spezialfall redundant realisierter Dateien (Multi-Kopien-Problem).

Als Zielsetzung dieser Arbeit wurde daher die Bereitstellung von dezentral organisierten Verfahren gewählt, die

- bei parallelen Zugriffen unterschiedlicher Transaktionen die operationale Integrität in einem zuverlässigen Gesamtsystem sichern,
- bei Störsituationen die Koordinierung der Maßnahmen des Systems zum Übergang in ein funktionsfähiges Restsystem und zur Fortführung eines geordneten Betriebs im Restsystem gewährleisten,
- die Koordination der Wiedereingliederung reparierter Komponenten in ein funktionsfähiges System sicherstellen.

Als Kern dieser Verfahren waren Kommunikationsprotokolle zu erarbeiten, die die Aktivitäten der Kontrollinstanzen aufgabengerecht koordinieren.

Aus dem breiten Spektrum der Aufgabenbereiche - Sperrung von Konstituenten, globale Datensicherung, Ausfall und Wiedereingliederung von Konstituenten und Kontrollinstanzen - ließen sich die Aufgabenbereiche

- Transaktionsbearbeitung bei zuverlässigem Gesamtsystem,
- Behandlung des Ausfalls von Komponenten der Datenbasis bei verbleibender Funktionsfähigkeit der Kontrollinstanzen,
- Behandlung des Ausfalls von Kontrollinstanzen,

als eigenständige Koordinierungsebenen isolieren und in einer Hierarchie anordnen. Es wurde gezeigt, wie alle o.a. Aufgabenbereiche in diese drei Koordinierungsebenen eingebettet werden können.

Neben dieser für die Lösung der gestellten Aufgabe wichtigen Strukturierung der Problematik konnte zudem aufgezeigt werden, daß für die Kommunikationsprotokolle der verschiedenen Koordinierungsebenen dasselbe Konzept anwendbar ist. Das gemeinsame Konzept ist das sog. Basisprotokoll, das sich zur Koordinierung beliebiger Aktivitäten von Kontrollinstanzen eignet.

Auf der Grundlage der eingeführten Hierarchie von Koordinierungsebenen und der Anwendung der Prinzipien des Basisprotokolls in jeder Koordinierungsebene wurden für den Übergang zwischen den Koordinierungsebenen die Regeln entwickelt, die eine verklemmungsfreie Kooperation der Kontrollinstanzen gewährleisten.

Das Basisprotokoll entstand durch Entsemantisierung eines der bekannten Verfahren zur Lösung des Multi-Kopien-Problems bei dezentralisierter Kontrolle; ein Vergleich der Leistungsfähigkeit existierender Lösungsansätze führte zu seiner Auswahl. Unterschiedliche Varianten des Basisprotokolls wurden einem Korrektheitsnachweis anhand eines auf Lindenmayer-Systemen basierenden formalen Modells unterzogen.

Auf der Grundlage des Basisprotokolls und der für zentralisierte Datenbanken entwickelten Sperrmechanismen wurden mehrere Verfahren entwickelt, die bei zuverlässigem Gesamtsystem die operationale Integrität der verteilten Datenbasis sichern.

Die Bedeutung der Verfahren liegt darin, daß sie zu ihrer Funktionsfähigkeit keine Einschränkungen bezüglich der Struktur und der Topologie der Datenbasis sowie der Komplexität des Transaktionsaufbaus voraussetzen und folgende Randbedingungen berücksichtigen:

- die koordinierte Ausführung von Transaktionen durch mehrere Kontrollinstanzen,
- die Koordinierung von Teilmengen von Kontrollinstanzen und damit die parallele Bearbeitung unabhängiger Transaktionen,
- die Anforderungen unterschiedlicher Sperrmechanismen.

Es wurden Verfahren mit fester und loser Kopplung von Kontrollinstanzen bei unterschiedlichem Aufwand bzgl. zu führender Information über den globalen Zustand der verteilten Datenbasis und bzgl. des Datenaustausches erarbeitet.

Für jedes Verfahren lassen sich Einsatzbereiche abgrenzen, in denen sein Koordinierungsaufwand im Vergleich zu dem der anderen Verfahren minimal ist. Weiterführende Untersuchungen des Leistungsverhaltens der Verfahren sind hier jedoch notwendig, um insbesondere dem in letzter Zeit stark wachsenden Bedarf an verteilten DV-Systemen im Bereich Prozeßlenkung durch die Bereitstellung systemtechnischer Planungshilfsmittel Rechnung zu tragen.

In dieser Ebene konnte auch die Koordinierung der Aktivitäten der Kontrollinstanzen zur globalen Datensicherung angesiedelt werden.

Die Behandlung des Ausfalls von Konstituenten und Kontrollinstanzen in übergeordneten Koordinierungsebenen war durch die Forderung bedingt, daß zur Sicherung der Integrität des Gesamtsystems die Kontrollinstanzen im Störungsfall unmittelbar reagieren müssen. Für den Einsatz in den unterschiedlichen Koordinierungsebenen der Ausfallbehandlung wurde das Basisprotokoll entsprechend modifiziert.

Die Wiedereingliederung von Konstituenten und Kontrollinstanzen wurde in die Koordinierungsebenen der o.a. Hierarchie eingearbeitet.

Zur Diskussion der Korrektheit der verklemmungsfreien Kooperation der Kontrollinstanzen unter Berücksichtigung der verschiedenen Koordinierungsebenen wurden entsprechende Erweiterungen des formalen Modells benutzt.

Die Integration der Konrollinstanzen als Prozesse in die Arbeitsrechner eines verteilten DV-Systems wurde unter Benutzung von

Funktionen des Nachrichtentransportsystems zur Interprozeßkommunikation aufgezeigt. Eine alternative Realisierungsvariante für Kontrollinstanzen ergibt sich, falls Koordinationsmechanismen teilweise oder ganz in spezielle Hardware wie z.B. Kommunikationsrechner ausgelagert werden können (Sekretärstechnik).

Zur Effizienzuntersuchung wurden für das Basisprotokoll beide Realisierungsvarianten als Simulationsmodelle implementiert. Um validierbare Aussagen zu gewinnen, wurde ein realitätsnahes Modell des Nachrichtentransportsystems eines existierenden Rechnernetzes verwendet. Die Experimente zeigten eine generelle Überlegenheit der Sekretärstechnik sowohl hinsichtlich der Belastung des Rechnernetzes durch das Basisprotokoll als auch der Festlegung optimaler Zeitschranken für die Überwachung der Aktionen der Kontrollinstanzen, um deren blockierungsfreie Kooperation zu gewährleisten.

Für verteilte Datenbanken existieren noch keine Architekturmodelle, auf die für eine Diskussion der Einbettung der Zugriffskontrolle in das Datenbankverwaltungssystem zurückgegriffen werden könnte. Der ANSI-SPARC-Report /A3/ schlägt für zentralisierte Datenbanken eine Gliederung des Architekturmodells in Schichten zur Realisierung folgender Schemata vor (vgl. /M2/):

- das externe Schema zur Unterstützung unterschiedlicher Benutzersichten der Datenbasis,
- das konzeptuelle Schema zur Gesamtsicht der in der Datenbank zu führenden umweltrelevanten Information,
- das interne Schema zur Aufnahme der physischen Aspekte der Abspeicherung der zur Datenbasis gehörenden Daten.

Ausgehend von diesem Architekturmodell lassen sich in jeder Schicht Verteilungsaspekte berücksichtigen. Eine denkbare Variante bietet sich durch Einführung einer zusätzlichen Schicht als Bindeglied zwischen der Ebene des internen Schemas und den lokalen Dateiverwaltungssystemen der Rechner im verteilten DV-System. Diese Schicht soll die wesentlichen mit der Verteilung von Datenbasen zusammenhängenden Aufgaben aufnehmen wie z.B. Führung von Kopien oder die Kontrolle des Zugriffs auf strukturell abhängige Konstituenten. Der Vorteil einer solchen Lösung würde vor allem in der Sicherung einer weitgehenden Unabhängigkeit des internen Schemas von der

Topologie und Kontrollstruktur der verteilten Datenbank bestehen.

Die in den vorgestellten Verfahren verwendeten fehlertoleranten Koordinierungsmechanismen können nicht nur zur koordinierten Betriebsmittelzuweisung in verteilten Datenbasen, sondern auch allgemein zur koordinierten Zuweisung wiederverwendbarer Betriebsmittel (resource sharing) in verteilten DV-Systemen eingesetzt werden (zu einem Beispiel aus dem Fertigungsbereich siehe /D7/). Dies beruht auf der Leistungsfähigkeit des Basisprotokolls, das die verklemmungsfreie Koordinierung beliebiger Aktivitäten der Kontrollinstanzen innerhalb einer Koordinierungsebene gewährleistet.

Die Sekretärstechnik in Verbindung mit neueren Systemtechnologien läßt sich auch hier sehr vorteilhaft anwenden. Zum einen ist die durch sie bedingte Reduktion der Belastung des verteilten DV-Systems insbesondere für den Einsatz des Basisprotokolls in verteilten DV-Systemen mit Kleinrechnern von Bedeutung; andererseits führt die mit ihr verbundene Modularisierung zur Abgrenzung derjenigen Aufgabenbereiche, die von einer Standardsoftware für dezentralisierte Betriebsmittelzuweisung in verteilten DV-Systemen wahrzunehmen sind. Eine solche Standardsoftware bietet insbesondere Planungsvorteile für die Realisierung inhomogener verteilter DV-Systeme.

Schlußwort

Die vorgestellten Verfahren zur Sicherung der operationalen Integrität von verteilten Datenbasen wurden im Rahmen eines übergeordneten Forschungsprojekts erarbeitet, das die Entwicklung und Implementierung eines Verwaltungssystems für verteilte Datenbasen in Kleinrechnernetzen zum Ziele hat. Das Forschungsprojekt wurde zum Teil vom Bundesministerium für Forschung und Technologie gefördert.

An dieser Stelle möchte ich Herrn Prof. Dr. G. Krüger besonders danken, der mir die Anfertigung dieser Arbeit ermöglichte und sie mit wertvollen Ratschlägen wesentlich unterstützte.

Herrn Prof. Dr. P.C. Lockemann danke ich für wichtige Hinweise und kritische Anmerkungen insbesondere bzgl. Fragen der Datenbanktechnologie.

Herrn Dr. E. Holler möchte ich besonders für seine wertvollen Anregungen und Impulse zur Lösung von Teilproblemen danken.

Herrn Dipl.-Math. B. Wolfinger danke ich für seine Mithilfe bei der Implementierung des Simulationsmodells für das Nachrichtentransportsystem und Herrn H. Marker für seine Unterstützung bei der Durchführung der Experimente.

Gleichfalls möchte ich mich bei den Mitarbeitern der Forschungsgruppe "Prozeßlenkung mit Mehrrechnersystemen" am IDT für ihre Diskussionsbeiträge danken, die diese Arbeit mitbeeinflußt haben.

Literaturverzeichnis

/A1/ Akkoyunlu, E., Bernstein, A., Schantz, R.
Interprocess communication facilities for network operating systems
Computer, June 1974, p. 46

/A2/ Alsberg, P.A., Day, J.D.
A principle for resilient sharing of distributed resources
Proc. 2nd International Conference on Software Engineering
San Francisco, California, Oct. 1976
IEEE Catalog No. 76CH1125-4C

/A3/ ANSI/X3/SPARC
Study Group on Data Base Management Systems
Interim Report 1975
Doc. No. 7514TS01

/A4/ Avizienis, A.
Fault-tolerant systems
IEEE Transactions on Computers, Vol. C-25, No. 12, December 1976

/B1/ Bayer, R.
On the integrity of databases and resource locking
Informatik-Symp. IBM Deutschland, Bad Homburg, Sept. 1975

/B2/ Bayer, R.
Integrity, concurrency, and recovery in databases
ECI 1976, Springer Verlag, Lecture Notes 44

/B3/ Bell, C.G., Habermann, A.N., McCredie, J., Rutledge, R., Wulf, W.
Computer Networks
Computer Science Review, 1969, Carnegie Mellon Univ.

/B4/ Bjork, C.A.
Recovery semantics for a DB/DC system
Proceedings ACM National Conference, p. 142-146

/C1/ Cerf, V., McKenzie, A., Scantlebury, R., Zimmermann, H.
Proposal for an international end-to-end protocol
SIGCOMM, Jan. 1976, Vol. 6, No. 1

/C2/ Chamberlin, D.D., Boyce, R.F., Traiger, I.L.
A Deadlock-free Scheme for Resource Locking in a Data Base Environment
Information Processing 1974, p. 340-343

/C3/ Chandy, K.M.
A survey of analytic models of rollback and recovery strategies
Computer, May 1975, p. 40

/C4/ Codd, E.F.
A relational model of data for large shared data banks
CACM, Vol. 13, No. 6, June 1970, p. 377

/C5/ Coffman Jr., E.G., Denning, P.J.
Operating Systems Theory
Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1973

/D1/ Dahl, O.J., Myhrhaug, B., Nygaard, K.
SIMULA 67 - Common Base Language
Norwegian Computing Center, Forskningsveien 1B, Oslo, 1968

/D2/ Date, C.J.
An Introduction to Database Systems
Addison-Wesley Publishing Company 1975

/D3/ Davenport, R.A.
Database integrity
The Computer Journal, Vol. 19, No. 2, May 1976

/D4/ Dijkstra, E.W.
Co-operating Sequential Processes
in: F. Genuys (ed.) Programming Languages, Academic Press, 1968

/D5/ Dijkstra, E.W.
Hierarchical ordering of sequential processes
Operating Systems Techniques, Academic Press, 1972

/D6/ Drobnik, O., Schumacher, F., Senger, R.
Simulation von Warteschlangensystemen:
Programmsystem zur Stichprobenerhebung und -auswertung
Ges. für Kernforschung Karlsruhe, KFK 1979, Karlsruhe, 1974

/D7/ Drobnik, O.
Strukturmodelle dezentralisierter Kontrolle in Mehrrechnersystemen
NTG-GI-Fachtagung "Rechnernetze und Datenfernverarbeitung", Aachen, 1976, Informatik Fachberichte 3, Springer-Verlag

/E1/ Ellis, C.A.
The duplicate database problem
1976, unveröffentlicht

/E2/ Eswaran, K.P., Gray, J.N., Lorie, R.A., Traiger, I.L.
On the Notions of Consistency and Predicate Locks in a Data Base System
IBM Research Report RJ 1487, Dec. 30, 1974
siehe auch: CACM, Nov. 1976, Vol. 19, No. 11, p. 624

/E3/ Everest, G.C.
Concurrent Update Control and Data Base Integrity
in: Data Base Management (ed. Klimbie, J.W., and Koffeman, K.L.), North Holland 1974, p. 241-270

/F1/ Forsdick, H.C.
A comparison of two schemes that control multiple updating of data bases
1975, unveröffentlicht

/G1/ Gouda, M.G., Manning, E.G.
On the modelling, analysis and design of protocols - a special class of software structures
Proc. 2nd International Conference on Software Engineering, San Francisco, California, Oct. 1976, IEEE Catalog No. 76CH1125-4C

/G2/ Gray, J.N., Lorie, R.A., Putzolu, G.R.
Granularity of Locks in a Shared Data Base
Proc. of the Intern. Conf. on Very Large Data Bases, 1975

/G3/ Gray, J.N., Lorie, R.A., Putzolu, G.R., Traiger, I.L.
Granularity of locks and degrees of consistency in a shared data base
GMD-Sommerseminar über Datenbanktechnologie, Bonn-St. Augustin, Juli 1976

/H1/ High Level Data Link Control Procedures
Proposed Draft International Standard on Elements of Procedures
International Organization for Standardization
ISO/TC 97/SC 6, Juli 1976

/H2/ Holler, E., Drobnik, O.
Modeling Network Control
GMD-GI-Proceedings of the European Workshop "Distributed Computer Systems", Darmstadt, October 1974

/H3/ Holler, E., Drobnik, O., Knöpker, R.
Entwurf und Modellierung von Mehrrechnersystemen für Prozeßlenkungsaufgaben
PDV-Bericht, Ges. für Kernforschung Karlsruhe, KFK-PDV 57, 1975

/H4/ Holler, E., Drobnik, O.
Rechnernetze
Bibliographisches Institut - Wissenschaftsverlag, Reihe Informatik/17, Mannheim/Wien/Zürich, 1975

/H5/ Holler, E., Krieger, J., Knöpker, R.
Ein universeller Kommunikationsprozessor für den Aufbau verteilter PDV-Systeme
2. Fachtagung "Prozeßrechner 1977", Augsburg, März 1977

/H6/ Holler, E.
Koordination kritischer Zugriffe auf verteilte Datenbanken in Rechnernetzen bei dezentraler Überwachung
Dissertation, Universität Karlsruhe 1974

/H7/ Holler, E., Drobnik, O.
Integrität, Ausfall und Wiederanlauf redundanter Prozeßdatenbasen in verteilten PDV-Systemen
2. Fachtagung "Prozeßrechner 1977", Augsburg, März 1977

/J1/ Johnson, P.R., Thomas, R.H.
The maintenance of duplicate databases
Network Working Group, RFC # 677, NIC # 31507, January 27, 1975

/K1/ King, P.F., Collmeyer, A.J.
Database Sharing - an Efficient Mechanism for Supporting Concurrent Processes
AFIPS Nat. Comp. Conf. Proc. 1973, p. 271-275

/K2/ Kirstein, P.T.
Developments in European Public Data Networks
GI-NTG-Fachtagung "Rechnernetze und Datenfernverarbeitung", Aachen 1976, Informatik-Fachberichte 3, Springer-Verlag, 1976, p. 39

/K3/ Kleene, S.C.
Introduction to Metamathematics
North-Holland Publishing Co. Amsterdam, 1962

/K4/ Krampe, Kubat, Runge
Bedienungsmodelle
Oldenbourg-Verlag München/Wien 1973

/L1/ Lampson, B., Sturgis, H.
Crash recovery in a distributed data storage system
Xerox, Palo Alto Research Center, 1976

/L2/ Le Moli
A Theory of Colloquies
IRIA Proceedings of the 1st European Workshop on Computer Networks, Arles, 1973

/L3/ Lockemann, P.C.
Informationssysteme
Vorlesungsskript, Universität Karlsruhe, 1974

/L4/ Lorie, R.A.
Physical integrity in a large segmented data base
IBM Research Report RJ 1767, April 1976

/M1/ Martin, J.
Security, accuracy and privacy in computer systems
Prentice-Hall, Inc., Englewood Cliffs, New Jersey 1973

/M2/ Martin, J.
Computer Data-Base Organization
Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1975

/M3/ Meister, B. et al.
On the Optimization of Message Switching Networks
IEEE Transactions on Communications, Vol. COM-20, No. 1, Febr. 1972, p. 8-14

/M4/ Moster, H.J.
Lösung des "Reader-Writer" Problems für Datenbasen in Rechnernetzen
Diplomarbeit, Universität Karlsruhe, Institut für Informatik III, 1976

/M5/ Mullery, A.P.
The distributed control of multiple copies of data
IBM- Research Report, Yorktown Heights, RC 5782 (# 25063)
August 1975

/P1/ Pouzin, L.
Standards in Data Communications and Computer Networks
4th Data Comm. Symp. 1975

/R1/ Rozenberg, G.
Theory of L-Systems: From the point of view of formal language theory
Lecture Notes in Computer Science 15, "L-Systems",
Springer-Verlag, 1974

/S1/ Sayani, H.H.
Restart and recovery in a transaction - oriented information processing system
Proc. ACM Sigfidet - Workshop, Ann Arbor, Michigan, May 1974

/S2/ Schlageter, G.
Access Synchronization and Deadlock-Analysis in Database Systems: An Implementation-Oriented Approach
Information Processing 1975, Vol. 1, p. 97

/S3/ Schlageter, G.
Konsistenzbedingungen für Datenbanksysteme: ein neues Problem der Systemanalyse
Angewandte Informatik 4/76, p. 159

/S4/ Senger, R.
Auftrags-Wartezeiten als Maß für die Bedienungsqualität eines Rechnerverbundnetzes
Diplomarbeit, Universität Karlsruhe, Dez. 1972

/S5/ Strack-Zimmermann, H.W., Schrödter, H.D.
The Hahn-Meitner-Institut Computer Network
NTG-GI-Fachtagung "Rechnernetze und Datenfernverarbeitung", Aachen 1976, Informatik Fachberichte 3, Springer-Verlag 1976

/S6/ Staudinger, W.
Datenübertragung in öffentlichen Nachrichtennetzen am Beispiel des öffentlichen Fernschreib- und Datennetzes der Deutschen Bundespost
NTG-GI-Fachtagung "Rechnernetze und Datenfernverarbeitung", Aachen 1976, Informatik Fachberichte 3, Springer-Verlag, 1976, p. 19

/T1/ Tanner, F.
Synchronisation in Datenbanken
Diplomarbeit, Universität Karlsruhe, Institut für Informatik II, 1976

/T2/ Thomas, R.H.
A solution to the update problem for multiple copy databases which uses distributed control
Bolt Beranek and Newman, Inc., 50 Moulton St., Cambridge, Mass., 02138, 1976

/W1/ Wilkes, M.V.
On preserving the integrity of data bases
The Computer Journal, Vol. 15, No. 3, 1972