UNIVERSITÄT LEIPZIG

Institut für Produktionswirtschaft
und Industrielle Informationswirtschaft

Marschnerstraße 31, 04109 Leipzig
Tel.: 0341/4941-182, Fax: -125

Arbeitsbericht Nr. 9

Petrinetzbasierte Modellierung komplexer Produktionssysteme

Band 5: Einführung in Synthetische Netze
Teilband 5.1: Darstellung des Kernkonzepts

von
Univ.-Prof. Dr. Stephan Zelewski

<zelewski @ hpswifa.wifa.uni-leipzig.de>

Leipzig 1995

Inhaltsverzeichnis zu Band 5.1

5 Synthetische Netze als elaborierte Version des Petrinetz-Konzepts

5.1 Das Kernkonzept Synthetischer Netze

5.1.1 Strukturanreicherung der Marken

5.1.1.1 Einführung in das Konzept sortierter Marken

Seine bedeutsamste[1] Erweiterung erfährt das Modellierungspotential des Petrinetz-Konzepts durch die Einführung strukturierter Marken[2]. Die zentrale Rolle dieser Strukturanreicherung äußert sich einerseits auf der formalen Betrachtungsebene. Dort bewirkt der Übergang von arithmetisch formulierten Stelle/Transition-Netzen zu algebraisch oder prädikatenlogisch definierten Höheren Netzen einen drastischen Komplexionssprung[3]. Andererseits wirkt sich die Strukturanreicherung von Marken ebenso auf der Ebene aus, auf der es um die Modellierung von Realitätsausschnitten geht. Aus dieser materialen Perspektive erlauben strukturierte Marken eine beträchtliche Steigerung der Modellierungsfähigkeit von Petrinetzen. Die Implikationen des Komplexionssprungs, den strukturierte Marken in Höheren Netzen ermöglichen, werden in den späteren Kapiteln anhand des Kernkonzepts Synthetischer Netze und seiner Erweiterungen ausführlich dargelegt. In diesem Kapitel wird zunächst nur der Aspekt einer strukturell reichhaltigeren Realitätsmodellierung vertieft.

Die Vergrößerung der Modellierungsfähigkeit beruht im wesentlichen auf der Möglichkeit, bei der Repräsentation von beweglichen Objekten nicht mehr auf die strukturlosen und ununterscheidbaren Marken von Stelle/Transition-Netzen eingeschränkt zu sein[4]. Vielmehr kann beweglichen Objekten, die durch Marken[5] repräsentiert werden, nunmehr auch eine differenzierte innere Struktur zugeschrieben werden[6]. Dieser Sachverhalt wirkt sich in dreifacher Weise aus. Erstens läßt sich mit unterschiedlich strukturierten Marken eine Vielfalt verschiedenartiger Objekte repräsentieren. Dies ist mit den Marken von Stelle/Transition-Netzen aufgrund ihrer Strukturlosigkeit grundsätzlich nicht möglich. Zweitens führt die Ununterscheidbarkeit von Marken in Stelle/Transition-Netzen dazu, daß dort nur individualitätslose Objekte dargestellt werden können. Strukturierte Marken lassen sich dagegen so weit ausdifferenzieren, daß es möglich wird, die repräsentierten Objekte in verschiedenen Modellzuständen als mit sich selbst identisch wiederzuerkennen[7]. Solche Marken stellen Individuen dar. Drittens können Veränderungen desselben Objekts im Zeitablauf als Modifizierungen von Markeneigenschaften modelliert werden. Dies setzt eine innere Markenstruktur voraus, um die veränderlichen Objekteigenschaften darzustellen[8].

Strukturierte Marken werden mit der strukturlosen Marke, die für Stelle/Transition-Netze definiert wurde, unter den Oberbegriff der sortierten Marken zusammengefaßt[9]. Sortierte Marken lassen sich in induktiver Weise einführen und zu beliebig komplexen formalen[10] Objekten ausgestalten.

Die unterste Induktionsstufe bilden atomare Objekte ohne innere Struktur[11]. Sie stellen Kopien von genau einer struktur- oder "farblosen"[12] Marke dar. Diese Marke wird als unstrukturierte Marke oder Basismarke bezeichnet. Die Basismarke besitzt keine Eigenschaften, die sie auszeichnen. Infolgedessen können ihre Kopien nicht voneinander unterschieden und erst recht nicht als individuelle Objekte erkannt werden[13]. Sie sind auf die ontologische Fundamentalkategorie beschränkt, entweder zu existieren oder nicht vorhanden zu sein. Alle existenten Kopien der Basismarke sind infolge Unterschiedslosigkeit notwendig miteinander identisch[14]. In einem Netz darf sich unter jeder Markierung eine beliebig große, aber endliche Anzahl solcher Kopien der Basismarke befinden[15]. Sie können zwar gezählt[16], nicht aber zu Mengen wohlunterschiedener Elemente zusammengefaßt werden.

Die einfachste Variante der strukturierten Marken stellen die Attributmarken dar. Jede Attributmarke zeichnet sich durch eine endliche Menge[17)] von Eigenschaften aus. Diese Markeneigenschaften werden in synonymer Weise als Attribute bezeichnet. Die Struktur einer Attributmarke wird durch die Anzahl ihrer Attribute und die Bereiche zulässiger Attributausprägungen festgelegt. Trotz ihrer inneren Struktur wird jede Attributmarke als ein atomares Objekt behandelt[18)]. Bei Attributmarken handelt es sich daher um atomare strukturierte Marken. Diese atomaren strukturierten Marken werden zusammen mit der unstrukturierten Basismarke unter den Begriff der einfachen Marken subsumiert.

Aus Attributmarken lassen sich strukturierte Marken mit ständig wachsender Strukturkomplexität zusammensetzen. Diese zusammengesetzten strukturierten Marken heißen auch komplexe Marken oder Kompositmarken. Eine Kompositmarke kann sowohl aus atomaren Attributmarken als auch aus anderen, aber weniger komplex zusammengesetzten Kompositmarken bestehen. Als Grenzfall wird zugelassen, daß eine Kompositmarke aus genau einer Attributmarke aufgebaut ist[19)].

Auch von strukturierten Marken können in einem Netz unter jeder Markierung beliebig, aber endlich viele Kopien existieren. Diese Kopien müssen jedoch nicht notwendig identisch sein. Vielmehr können sich Exemplare derselben strukturierten Marke unter derselben Markierung durch verschiedene Ausprägungen der Markenattribute voneinander unterscheiden (synchrone Attributvarietät[20)]). Ebenso vermag die Kopie einer strukturierten Marke bei Markierungsveränderungen ihre Attributausprägungen zu wechseln (diachrone Attributvarietät). Durch diese zweidimensionale Varietät der Ausprägungen von Markenattributen wird ein reichhaltiges Ausdruckspotential für Eigenschaften modellierter Objekte und deren Veränderungen eröffnet.

Als Attribute von atomaren strukturierten Marken werden in dieser Arbeit klassifizierende, quantitative und identifizierende Attribute unterschieden. Klassifizierende Attribute werden in der Literatur zum Petrinetz-Konzept vielfach verwendet. Identifizierende und quantitative Attribute finden dagegen nur selten besondere Beachtung[21)]. Dennoch werden sie später intensiv genutzt, um eine realitätsnahe Modellierung von Maschinenbelegungen bei Flexiblen Fertigungssystemen zu erleichtern[22)].

Klassifizierende Markenattribute ermöglichen es, das Schaltverhalten von Netzen sowohl zu differenzieren als auch zu aggregieren. Dabei sind beide Optionen in der Regel miteinander verwoben. In der ersten Hinsicht läßt sich das Schalten einer Transition für Marken, deren klassifizierende Attribute verschiedene Ausprägungen aufweisen, entsprechend unterschiedlich gestalten. Der Aggregationsaspekt erstreckt sich hingegen auf die Zusammenfassung von Transitionen, die zwar für verschiedenartige Marken gelten, deren Schaltvorschriften sich aber auf ein gemeinsames Schaltschema zurückführen lassen. Solche Transitionen können zu einer einzigen Transition aggregiert werden, deren Schaltvorschrift von den Ausprägungen der klassifizierenden Markenattribute abhängt[23)]. Diese Aufspaltung der Schaltvorschrift in verschiedene Schaltmodi kann ebenso als eine Manifestation des zuerst angesprochenen Differenzierungsaspekts aufgefaßt werden.

Wenn klassifizierende Markenattribute herangezogen werden, um das Schaltverhalten von Transitionen zu differenzieren und zu aggregieren[24)], wird zumeist von "farbigen" oder "gefärbten" Marken gesprochen[25)]. Die Schaltvoraussetzungen und -wirkungen von Transitionen unterscheiden dann zwischen Marken unterschiedlicher Farbe[26)]. Auf diese Weise wird es möglich, selbst umfangreiche Modellierungsobjekte noch durch recht kompakte Netze zu repräsentieren. Dies gilt insbesondere dann, wenn die Verhaltensweisen der Modellierungsobjekte von Klassen ähnlicher, aber nicht identischer Prozesse geprägt werden[27)]. Jeder dieser Prozesse muß als Markenfluß in einem prozeßspezifischen Teilnetz dargestellt werden, solange nur die eine "farblose" oder "schwarze"[28)] Basismarke zugelassen ist. Sobald jedoch Markenfarben eingeführt werden, können die Teilnetze, die zu ähnlichen Prozessen gehören, zu einem einzigen Teilnetz zusammengefaltet werden[29)]. Jede dieser Markenfarben repräsentiert dann genau einen derjenigen ähnlichen Prozesse, die den zusammengefalteten prozeßspezifischen Teilnetzen zugrundelagen. Insofern kann von einem prozeßspezifischen Charakter der Markenfarben gesprochen werden.

Von der Auffassung darüber, welche Prozesse als ähnlich zu klassifizieren sind, wird die Art und das Ausmaß der Teilnetzzusammenfaltung bestimmt. Im Prinzip ist es bei Höheren Netzen mit gefärbten Marken immer möglich, den Ähnlichkeitsbegriff so weit auszulegen, daß er alle Prozesse umfaßt, die das Verhalten eines Modellierungsobjekts determinieren. In diesem Extremfall nimmt die "zusammengefaltete" Netzrepräsentation des Modellierungsobjekts die Gestalt eines vollständig kompaktifizierten Netzes an: Es besteht nur noch aus genau einer Stelle und genau einer Transition, die durch mindestens eine adjazente Kante miteinander verknüpft sind[30]. Die Verhaltenskomplexität des repräsentierten Modellierungsobjekts ist dann aus der simplen Topologie des Kompaktnetzes nicht mehr ersichtlich. Statt dessen ist sie in der großen Anzahl der erforderlichen Markenfarben[31] und in der komplizierten, von jenen Markenfarben abhängigen Schaltregel der einen Transition verborgen[32]. Netzmodelle werden in dieser Arbeit u.a. untersucht, weil die Visualisierung von netzrepräsentierenden Graphen eine hohe Anschaulichkeit der Modellierung verspricht[33]. Daher wird darauf verzichtet, die vollständige Kompaktifizierung Höherer Netze zu verwirklichen - oder auch nur anzunähern. Vielmehr werden Markenfarben nur in dem Ausmaß eingesetzt, wie sie eine zwar kompakte, aber dennoch weiterhin transparente Darstellung von Netzmodellen erlauben. Es erfolgt also nur ein "kontrollierter" Übergang von umfangreichen Netzmodellen, die mit der Hilfe von Niederen Netzen (Stelle/ Transition-Netzen) formuliert sind, zu kompakteren Netzmodellen, die das Ausdrucksvermögen von Höheren Netzen (Synthetischen Netzen) nutzen.

Klassifizierende Attribute wurden zuvor nur hinsichtlich ihrer herausragenden Funktion beleuchtet, in ihrer Verwendung als Markenfarben die Konstruktion kompakter Netzmodelle zu gestatten. Darüber hinaus können klassifizierende Markenattribute aber ebenso benutzt werden, um in einem Netzmodell den Informationswert der Repräsentation beweglicher Objekte zu erhöhen. Die Markenattribute erlauben dann, beliebige Eigenschaften der betroffenen Objekte auszudrücken. Dabei läßt es das Konzept strukturierter Marken sogar zu, die Objekteigenschaften durch eine verfeinernde Differenzierung von Subklassen hierarchisch zu systematisieren[34]. Diese Möglichkeit wird später im Rahmen der Fallstudie intensiv genutzt.

Durch die Verwendung quantitativer Markenattribute wird es möglich, mit der Hilfe von Petrinetzen auch numerische Eigenschaften von Objekten einfach und umfassend[35] zu modellieren. Dies stellt eine bedeutsame Erweiterung dar. Denn das Petrinetz-Konzept abstrahiert in seiner ursprünglichen, rein topologisch ausgerichteten Fassung der Stelle/Transition-Netze bewußt von allen metrischen Kategorien. Diese quantitative Strukturanreicherung ist in bezug auf die beabsichtigte Modellierung von Maschinenbelegungen bei Flexiblen Fertigungssystemen notwendig, da die Erfüllung der wesentlichen Formalziele - wie z.B. der Kapitalkosten- oder der Durchlaufzeitminimierung - auf metrischen Skalen gemessen wird. So lassen sich Auswirkungen der Auftragsausführung auf diese Formalziele nur unmittelbar erfassen, wenn den auftragsabbildenden Marken Beiträge zur Zielverwirklichung als numerische Attributausprägungen[36] zugeschrieben werden.

Einen letzte wesentliche Erweiterung der Modellierungsfähigkeit bringen identifizierende Markenattribute mit sich. Sie ermöglichen es, Aspekte eines Systems zu modellieren, die nicht nur von der Anzahl gleichartiger Objekte oder ihren gemeinsamen Eigenschaften abhängen. Vielmehr können nun auch individuelle Charakteristika einzelner Objekte repräsentiert werden. Insbesondere wird es auf diese Weise möglich, ein individuell bestimmtes Objekt an jedem Ort und zu jedem Zeitpunkt innerhalb eines Netzmodells als solches wiederzuerkennen. So wird z.B. eine Werkstückmarke mit einem identifizierenden Attribut für den Werkstücknamen eingeführt, um den Verbleib eines Werkstücks im Produktionssystem jederzeit feststellen und im Zeitablauf verfolgen zu können. Ein weiteres, jetzt aber klassifizierendes Attribut der Werkstückmarke gibt den Namen desjenigen Produktionsauftrags an, zu dem das Werkstück gehört. Das Zusammenspiel von identifizierendem Werkstücknamen und klassifizierendem Auftragsnamen gestattet es, im Netzmodell eines Produktionssystems eine differenzierte Auftragsverfolgung zu verwirklichen: Jederzeit läßt sich anhand der Markenverteilung im Netzmodell exakt feststellen, an welchem Ort sich die einzelnen Werkstücke eines vorgegebenen Produktionsauftrags befinden. Ebenso kann im Zeitablauf beobachtet werden, wie die auftragszugehörigen Werkstücke das modellierte Produktionssystem durchsetzen.

Die Einführung von Marken mit innerer Struktur erweitert das Ausdruckspotential von Petrinetzen nicht nur im Hinblick auf die differenzierte Darstellung von Objekteigenschaften. Vielmehr eröffnet sie auch neuartige Möglichkeiten, zeitliche Entwicklungen der jeweils modellierten Objekte in Netzmodellen abzubilden. Die strukturlosen Marken von Stelle/Transition-Netzen lassen die Darstellung von Objektveränderungen nur in zwei Dimensionen zu. Es ist einerseits möglich, die Anzahl von Marken auf Stellen zu verändern, ohne die Marken zu bewegen[37)]. Andererseits können die Marken so zwischen den Stellen ausgetauscht werden, daß die Markenverteilung im Netz verändert wird, ohne hierdurch ihre Anzahl im Gesamtnetz zu variieren[38)]. Diese beiden Modi lassen sich beliebig miteinander kombinieren[39)]. Strukturierte Marken eröffnen dagegen eine dritte, grundverschiedene Möglichkeit, Entwicklungen eines modellierten Objekts abzubilden[40)]. Trotz konstanter Markenanzahl und -verteilung lassen sich die Ausprägungen der Markenattribute modifizieren. Daher wird die Schaltregel Synthetischer Netze derart abgewandelt, daß sie auch die Veränderung der Ausprägungen von Markenattributen als Schaltwirkung zu spezifizieren gestattet.

Die strukturierten Marken lassen sich sowohl in das Konzept der objektorientierten Systemgestaltung als auch in konzeptuelle Schemata für Datenbanksysteme einordnen. Beide Bezüge klangen bereits im Kontext identifizierender Markenattribute an. Aus der Perspektive der objektorientierten Systemgestaltung[41)] stellt jede strukturierte Marke eine Objektklasse dar. Eine Markenkopie entspricht dagegen der Instantiierung einer Objektklasse durch ein einzelnes Objekt. Aus dem Blickwinkel konzeptueller Datenbankschemata[42)] entspricht eine strukturierte Marke einem Datentyp (Entity-Typ) beliebiger Komplexität. Jede Kopie einer solchen Marke ist ein konkretes Datenobjekt (Entity[43)]). Eine Attributmarke, die nur genau ein Attribut umfaßt, stellt ein einfaches Datum dar. Attributmarken aus mehreren Attributen lassen sich als lineare Listen auffassen. Zusammengesetzte Marken definieren hierarchisch organisierte Listen.

Mit der Hilfe zusammengesetzter Marken können komplexere Datentypen ausgedrückt werden, als es mit derzeit üblichen betriebswirtschaftlichen Datenschemata möglich ist. Diese Schemata für die Implementierung realer Datenbanksysteme beruhen - allenfalls - auf einer relationalen Datenbankarchitektur[44)]. Als relationaler Ansatz gestattet sie nur solche Datenobjekte zu typisieren, die lineare Listen darstellen[45)]. Auch wenn abstrakter formulierte konzeptuelle Datenschemata herangezogen werden, dominiert weiterhin der relationale Ansatz mit seiner listenförmigen Datentypen. So greift beispielsweise auch das Unternehmensdatenmodell, das derzeit von SCHEER als umfassendes konzeptuelles Schema für die betriebliche Informationswirtschaft propagiert wird[46)], auf einen relationalen Ansatz - das Entity-Relationship-Konzept[47)] - zurück[48)].

In der Ausrichtung konventioneller[49)] relationaler Schemata auf Datentypen, die nur die Gestalt linearer Listen anzunehmen vermögen, liegt jedoch eine erhebliche Einschränkung. Denn auf dieser Basis können nur "flache" Objekte unmittelbar dargestellt werden[50)]. Dies wird der Anforderung einer "natürlichen" Abbildung komplexer Objekte mit intern verschachtelter Struktur nicht gerecht[51)]. Ein *einzelnes* solches Objekt kann in einem relationalen Datenschema nur durch *mehrere* verknüpfte lineare Listen erfaßt werden[52)]. Eine natürliche Datenmodellierung läge aber erst dann vor, wenn es gelänge, ein verschachtelt strukturiertes Objekt durch genau ein Datenobjekt wiederzugeben[53)]. Genau dies leisten die Kopien von zusammengesetzten strukturierten Marken aus Synthetischen Netzen.

Abb. 31 gewährt einen Überblick über das Markenkonzept Synthetischer Netze. Da jede Markenkopie als ein Datenobjekt aufgefaßt werden kann, wurde der Konzeptüberblick in Anlehnung an JACKSON's Darstellungsweise datenstrukturorientierter Programmentwürfe gestaltet[54)]. Die Rekursivität der Markenontologie äußert sich darin, daß strukturierte Marken (höherer Komplexität) strukturierte Marken (niedrigerer Komplexität) als Komponenten enthalten können.

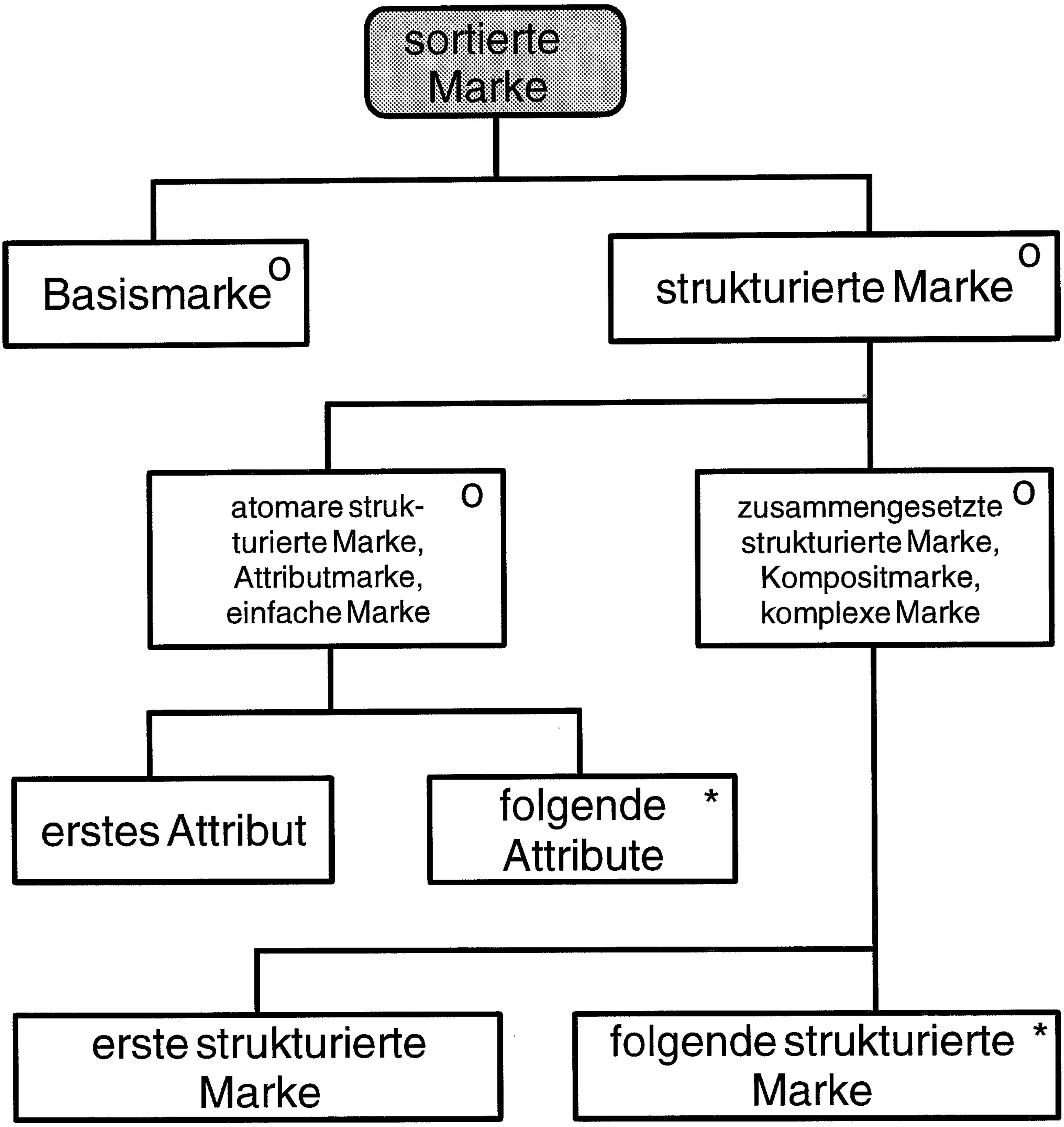

Abb. 31: Das Markenkonzept Synthetischer Netze

Anmerkungen zum Kapitel:

1) Vgl. zur Begründung dieses Urteils die zusammenfassende Würdigung der Einführung strukturierter Marken.

2) Die Bezeichnungsweise der nachfolgend beschriebenen Marken schwankt in der Netzliteratur. Der Verf. präferiert das Attribut "strukturiert", weil es die interne Markenzusammensetzung aus Komponenten unmittelbar ausdrückt. REISIG (1985d), S. 185ff.; VAUTHERIN (1987b), S. 293; REISIG (1989a), S. I, 1, 10 u. 41ff., und ABEL,D. (1990), S. 35 u. 37, sprechen dagegen von individuellen Marken. Diese Bezeichnung erscheint dem Verf. allerdings bedenklich. Denn die Marken können zwar identifizierende Attribute besitzen, müssen es aber keineswegs. Bei BATTISTON (1988), S. 20 u. 23, werden die Ausdrucksweisen strukturierter und individueller Marken miteinander gleichgesetzt.
Strukturierte Marken ähneln stark den "beweglichen Objekten", die von ESCHENBACHER (1989), S. 122, und ESCHENBACHER (1991), S. 224, 226f. u. 233, im Rahmen der Simulationssprache SIMPLEX-MDL vorgestellt werden. Diese beweglichen Simulationsobjekte entsprechen zunächst den Attributmarken, die in Kürze eingeführt werden. Darüber hinaus können die Objekte spezielle "Locations" umfassen, in denen ein bewegliches Objekt andere bewegliche Objekte mitzuführen vermag; vgl. ESCHENBACHER (1989), S. 122; ESCHENBACHER (1991), S. 226 u. 233. Dies kommt der Zusammensetzung strukturierter Marken in Kompositmarken gleich. Allerdings reicht das nachfolgend entfaltete Konzept strukturierter Marken über die beweglichen Objekte der Simulationssprache SIMPLEX-MDL hinaus. Denn die hierarchisch verschachtelten Attributstrukturen, die an späterer Stelle für Attributmarken eingeführt werden, finden in den beweglichen Simulationsobjekten keine Korrespondenz.

3) Der Komplexionssprung äußert sich mittelbar darin, daß sich Petrinetze bei der Verwendung von strukturierten Marken wesentlich kompakter formulieren lassen, als es bei einer Beschränkung auf unstrukturierte Marken möglich wäre. Vgl. zu dieser Kompaktifizierung von Petrinetzen durch Rückgriff auf strukturierte Marken FREEDMAN (1988b), S. 331; REISIG (1989a), S. 1 u. 42.
Umgekehrt stellt sich eine "kombinatorische Explosion" des Netzumfangs ein, wenn versucht wird, mit den unstrukturierten Marken der Niederen Netze auszukommen. Dies wird besonders deutlich bei FREEDMAN (1988b), S. 337 u. 339f. Dort müssen alle Zustandsvariablen eines Produktionsmodells, in so viele Bedingungen aufgespalten werden, wie die ursprünglichen Zustandsvariablen unterschiedliche Ausprägungen anzunehmen vermögen. Nur auf diese Weise ist es möglich, mit einem Bedingung/Ereignis-Netz auszukommen. In ihm zeigt die (Nicht-)Markierung einer variablen- *und* ausprägungsspezifischen Stelle an, ob die jeweils betroffene Ausprägung einer Zustandsvariablen (nicht) vorliegt. Darüber hinaus sind solche kombinatorischen Variablenaufspaltungen nur so lange möglich, wie die Menge zulässiger Variablenausprägungen endlich bleibt. Dies muß jedoch keineswegs immer der Fall sein.

4) Diesen Sachverhalt scheint ESCHENBACHER (1991), S. 223, vollkommen zu übersehen, wenn er dem Petrinetz-Konzept polemisch vorhält: "Die Marken, die in Petri-Netzen verschoben werden, suggerieren, daß diese bewegliche Teile repräsentieren könnten. Das ist aber nicht der Fall. Die Marken sind Zustandsgrößen, die den Wahrheitswert von Bedingungen anzeigen." Anscheinend hat ESCHENBACHER nur die Klasse der Bedingung/Ereignis-Netze zur Kenntnis genommen, auf die seine Feststellung tatsächlich zutrifft. Die Möglichkeit, Marken aus Stelle/Transition-Netzen als Repräsentanten von Ressourcen zu interpretieren, ignoriert er. Ebensowenig geht er auf die vielfältigen Ausdrucksmöglichkeiten ein, die strukturierte Marken eröffnen. Um Mißverständnisse der oben zitierten Art auszuräumen, wird in diesem Kapitel dem Modellierungspotential strukturierter Marken größere Aufmerksamkeit zuteil.

5) Strenggenommen werden die beweglichen Objekte nicht durch Marken, sondern durch die Kopien von Marken repräsentiert. Die Unterscheidung zwischen Marken und ihren Kopien wurde bereits im Kontext von Stelle/Transition-Netzen eingeführt. Diese Differenzierung spielte bei Stelle/Transition-Netzen jedoch noch keine Rolle, weil alle Kopien ihrer einen (Basis-)Marke identisch sind. Daher wurden dort die Markenkopien kurz als Marken angesprochen. Diese vereinfachte Redeweise wird weiterhin beibehalten, sofern die Unterscheidung zwischen Marken und Markenkopien den Argumentationsfluß behindern würde, ohne einen nennenswerten Erkenntnisgewinn zu vermitteln. Sobald die Differenzierung aber eine inhaltliche oder formale Bedeutung erlangt, wird sie entsprechend berücksichtigt.

6) REISIG (1989a), S. I u 1, hebt hervor, daß sich strukturierte Marken zur Repräsentation beliebig strukturierter Objekte eignen. Gleiches gelte für Datenstrukturen jeder Art. Auf die Beziehung zwischen Petrinetzen mit strukturierten Marken einerseits und Daten andererseits wird an späterer Stelle zurückgekommen.

7) Die epistemologische Problematik, wie ein Objekt trotz seiner Veränderungen als mit sich selbst identisch wiedererkannt werden könne, führt über den Erkenntnisrahmen dieser Arbeit hinaus. Beispielsweise ließe sich auf das "Prinzip der nominalen Invarianz" zurückgreifen. Es wird näher erläutert bei BUNGE (1977), S. 221, und WAND (1989), S. 545 (i.V.m. S. 544) sowie S. 549. In Synthetischen Netzen wird statt dessen pragmatisch verfahren: Ein identifizierendes Markenattribut konstituiert die Identität einer objektrepräsentierenden Marke(nkopie).

8) Darüber hinaus ist es ebenso erforderlich, die veränderten Marken als Repräsentanten desselben - identischen - Objekts wiederzuerkennen. Diese Facette wurde jedoch schon kurz zuvor angesprochen.

9) Hiermit wird die Rückbindung der Marken von Synthetischen Netzen an das algebraische Signaturkonzept auch taxonomisch verdeutlicht.

10) Mit dem Begriff "Objekt" ist hier das Formalobjekt gemeint, das die Marke selbst ist. Es ist deutlich von den voranstehend angeführten Objekten aus dem jeweils modellierten Realproblem zu unterscheiden, die als Realobjekte vom Formalobjekt "Marke" repräsentiert werden.

11) Das Fehlen einer Markenstruktur wird als "Nullstruktur" oder degenerierter Grenzfall des allgemeinen Konzepts strukturierter Marken aufgefaßt.

12) Diese Bezeichnung wird bei der späteren Einführung "gefärbter" Marken einsichtig. Der farblosen Marke m_0 läßt sich dann die Pseudo-"Farbe" Schwarz zuordnen.

13) Notwendige Voraussetzung für die Individualität eines Objekts ist eine nicht-leere Menge von Objekteigenschaften, bezüglich derer sich ein individuelles Objekt von allen anderen Objekten unterscheiden kann.

14) Dabei wird LEIBNIZ' Prinzip der Identität des Ununterscheidbaren vorausgesetzt.

15) Dies schließt das Vorhandensein keiner Marke ein.

16) Infolge dieser Zählbarkeit von unstrukturierten Marken(kopien) konnten die Markierungen von Stelle/Transition-Netzen in arithmetischer Weise als Funktionen auf natürlichzahligen Wertebereichen definiert werden.

17) Zunächst wird unterstellt, daß jede Attributmarke mindestens ein Attribut umfaßt. Später wird jedoch die Fallstudie zeigen, daß es für die Repräsentation von "Nichtobjekten" empfiehlt, als degenerierten Grenzfall auch "nullstellige" Attributmarken zuzulassen. Ihre Attributemenge ist daher zwar immer noch endlich, aber leer.

18) Ein atomares Objekt wird hier - analog zu den physikalischen Atomen mit einer komplexen inneren, von Elementarteilchen und Kraftfeldern geprägten Struktur - nicht notwendig als strukturlos betrachtet. Die Bezeichnung "atomar" drückt lediglich aus, daß es sich - im Rahmen des Netzkonzepts - um ein kleinstes selbständiges Objekt handelt, dessen Komponenten (sofern solche existieren) innerhalb dieses Konzepts nicht mehr als selbständige Objekte definiert sind. Atomare strukturierte Marken besitzen zwar eine innere Struktur. Doch können ihre Strukturkomponenten - die Markenattribute - nicht selbständig existieren. In dieser ontologischen Prämisse besteht ein wesentlicher Gegensatz zur Konzeption der Prädikat/Transition-Netze, in denen Attribute und Marken jeweils selbständig auftreten können und formal nicht streng auseinandergehalten werden. Auf diese "markenontologischen" Differenz zwischen Prädikat/Transition-Netzen und Synthetischen Netzen wurde bereits kurz hingewiesen.

19) Diese degenerierte "Zusammensetzung" erlaubt es, dasselbe reale Objekt - aus verschiedenen Perspektiven - einmal als Teil und einmal als Ganzes zu betrachten. Dies empfiehlt sich vor allem für die Modellierung von Produktstrukturen. Denn dasselbe Objekt - z.B. ein Getriebe - kann sowohl das Einbauteil eines komplexeren Produkts als auch ein selbständiges Produkt darstellen. Falls ein Produkt generell als Komplex seiner Bestandteile definiert wird und das Getriebe als ein atomarer Bestandteil modelliert wurde, liegt eine degenerierte Produktzusammensetzung aus genau einem Bestandteil vor.

20) Die Unterscheidung zwischen syn- und diachroner Attributvarietät lehnt sich an die Differenzierung zwischen syn- und diachroner Betrachtungsweise in den Sprachwissenschaften an; vgl. DE SAUSSURE (1967), S. 94ff., insbesondere S. 96, 106ff. u. 119.

21) Zwar benutzen etliche Autoren die Formulierung, ihre Netzklassen erfaßten individuelle Marken. Doch werden von diesen "individuellen" Marken mehrere Kopien unter derselben Markierung zugelassen. Dies widerspricht ihrer Definition als Individuen, sofern von dem speziellen Fall abgesehen werden, daß die verschiedenen Markenkopien unterschiedliche Sichtweisen der Modellgestalter oder -benutzer auf dasselbe Individuum darstellen sollen. Diese perspektivische Vervielfachung individueller Objekte wird von den vorgenannten Autoren jedoch nicht in Anspruch genommen. Dem Verf. sind keine Arbeiten zur Petrinetz-Theorie bekannt, die sich tatsächlich mit individuellen Marken im strengen Sinne des Individualitätsbegriffs befassen.
Im Hinblick auf quantitative Markenattribute existieren zwar vereinzelte Ansätze, insbesondere im Rahmen von netzbasierten Simulationskonzepten. Aber diese quantitativen Strukturanreicherungen vermochten sich bis heute noch nicht als "etabliert" durchzusetzen.

22) Vgl. hierzu die nachfolgend kurz angeführten Beispiele und die ausführlichen Netzmodule, die an späterer Stelle präsentiert werden.

23) Vgl. ABEL,D. (1990), S. 39f.

24) Die gleichen Aggregations- und Differenzierungseffekte können im Prinzip ebenso mit klassifizierenden Markenattributen erreicht werden, die einen quantitativen Charakter tragen. Diese Variante wird aber - wie schon oben angedeutet - in der Netzliteratur kaum beachtet.

25) Farbige Marken werden vorrangig durch die Klasse "Gefärbter Netze" erfaßt. Implizit finden farbige Marken auch in der Klasse der Prädikat/Transition-Netze Berücksichtigung; vgl. JENSEN (1987a), S. 249. Auf Prädikat/Transition-Netze wird in Kürze näher eingegangen.

26) Dafür müssen die Schaltregeln entsprechend farbabhängig modifiziert werden. Diese Modifizierung erfolgt durch Definition farbabhängiger Schaltmodi der Transitionen. Solche Schaltmodi werden vom Konzept der Operationsmodi, das für das allgemeine Übergangsschema von Synthetischen Netzen eingeführt wird, als Spezialfall umgriffen. Später werden die Schaltmodi oder "Schaltfarben" von Transitionen aus Synthetischen Netzen ausführlicher erläutert. Vgl. ebenso die anschauliche Erklärung von farbabhängigen Schaltmodi für die Transitionen eines Gefärbten Netzes bei ABEL,D. (1990), S. 39f.

27) Vgl. JENSEN (1987a), S. 249 u. 251ff.; ABEL,D. (1990), S. 38 u. 41f. (er spricht von gleichartigen Teilsystemen, meint aber inhaltlich die oben angesprochenen Prozesse).

28) Vgl. ABEL,D. (1990), S. 37.

29) Vgl. zu Beispielen für solche Netzfaltungen REISIG (1989b), S. 5 (i.V.m. mit dem "Cover"); ABEL,D. (1990), S. 35ff.
Ob die Teilnetzfaltungen entweder schrittweise oder aber uno actu erfolgen, ist im Prinzip unerheblich. Hier wird der Einfachheit halber eine Faltung unterstellt, die in nur einem Schritt geschieht. Dem ausführlicheren Beispiel, das in der anschließenden Anmerkung erläutert wird, liegt dagegen eine sukzessive Faltung zugrunde.

30) Vgl. zu dieser vollständigen Netzkompaktifizierung ABEL,D. (1990), S. 42.
Dies Möglichkeit einer vollständigen Netzkompaktifizierung wird anhand eines Beispiels verdeutlicht. Es dient zugleich dazu, den oben erwähnten Zusammenhang zwischen Prozessen, prozeßspezifischen Teilnetzen und daraus abgeleiteten Markenfarben zu verdeutlichen. Das Beispiel entstammt in seinen Grundzügen einer ausführlichen Studie von ABEL,D. (1990), S. 35ff. (i.V.m. S. 22ff.). Es beschäftigt sich mit zwei Robotern, die Werkstücke auf zwei Fertigungslinien "A" und "B" handhaben. Als Besonderheit kommt hinzu, daß für die Handhabung eines Werkstücks von der Fertigungslinie "A" beide Roboter zusammenwirken müssen. Bei der Handhabung eines Werkstücks von der Fertigungslinie "B" ist dagegen nur einer von den beiden Robotern erforderlich. Um einen Vergleich mit ABEL's Modellierungsvorschlag nicht unnötig zu erschweren, wird seine Notation im folgenden weitgehend übernommen. Das gilt insbesondere für seine Großschreibung von Konstanten und seine Kleinschreibung von Variablen. (Ansonsten wird in dieser Arbeit gemäß der PROLOG-Konventionen die umgekehrte Notation gepflegt.) Gleiches trifft auf die natürlichsprachlichen Knotenanschriften zu. Die Indizes der Netzknoten entsprechen der Knotennumerierung bei ABEL.
Allerdings wird von ABEL's Modellierung im folgenden in zweifacher Hinsicht abgewichen. Erstens hat ABEL eine vollständige Kompaktifizierung seines Netzmodells unterlassen. Dafür hat er keine Gründe angegeben. Zweitens wird hier von einem Stelle/Transition-Netz ausgegangen, das gegenüber dem Stelle/Transition-Netz der Abb. 3.9 bei ABEL,D. (1990), S. 36, leicht modifiziert ist. Denn ABEL läßt zu, daß die Roboter "I" und "II" in seinem Netz während des Schalten seiner ersten oder zweiten Transition vorübergehend verschwinden und später beim Schalten seiner dritten oder vierten Transitionen wieder auftauchen können. Dies widerspricht einer "natürlichen" Modellierung. Darüber hinaus resultiert daraus eine unvollständige Erfassung derjenigen Prozesse, die das Gesamtverhalten des modellierten Realitätsausschnitts prägen. Um diese Prozesse durch Markenfarben vollständig zu erfassen, wird das Ausgangsnetz hier in einer entsprechend erweiterten Version vorgelegt. Die anschließende Kompaktifizierung des modifizierten Stelle/Transition-Netzes weicht aufgrund dieser Ergänzungen von ABEL's Vorgehen in Details ab. Die grundsätzliche Stoßrichtung der schrittweisen Netzkompaktifizierung wird dadurch aber nicht verändert.
Insbesondere übernimmt der Verf. von ABEL den Gedanken, die sukzessiv kompakteren Höheren Netze einerseits als Gefärbte Netze und andererseits als Prädikat/Transition-Netze zu präsentieren. Beide Netzvarianten stimmen zwar auf jeder Kompaktifizierungsstufe hinsichtlich ihrer gemeinsamen Netzgraphik überein. Aber die Inzidenzmatrizen der Gefärbten Netze und der Prädikat/Transition-Netze fallen deutlich auseinander. Die Prädikat/Transition-Netze werden hier berücksichtigt, weil diese Netzklasse in dieser Arbeit immer wieder als "typischer" Vertreter für Höhere Netze angesprochen wird. Die Gefärbten Netze werden ebenso angeführt, weil sie im hier erörterten Kontext die Ableitung von Markenfarben aus Prozessen des Modellierungsobjekts besonders deutlich lassen. Es wird jedoch darauf verzichtet, Aufbau und Schaltverhalten von Gefärbten Netzen und Prädikat/Transition-Netzen detailliert zu erklären. Denn sie gehören nicht zum engeren Erkenntnisinteresse der hier vorgelegten Ausarbeitung. Für das Verständnis des nachfolgend präsentierten Beispiels reichen die Grundzüge des Petrinetz-Konzepts aus, die zuvor anhand von Stelle/Transition-Netzen erarbeitet wurden.
Ausgangspunkt der Kompaktifizierung ist das Stelle/Transition-Netz aus Abb. 32. Es modelliert den oben erwähnten Zugriff zweier Roboter auf zwei Fertigungslinien. Das Verhalten dieses Miniatur-Produktionssystems wird durch vier Prozesse geprägt: je einen zyklischen Werkstückfluß auf der Fertigungslinie "A" und "B" sowie je einen Hand-

habungsprozeß für den Roboter "I" und "II". Diese vier Prozesse sind in der Abb. 32 durch entsprechende Schraffuren und Umrandungen hervorgehoben. Die Ausgangsmarkierung des Stelle/Transition-Netzes mit Kopien der "schwarzen" Basismarke zeigt an, daß 4 Werkstücke am Beginn der Fertigungslinie "A" und 3 Werkstücke am Beginn der Fertigungslinie "B" warten. Hinzu kommt je eine Basismarkenkopie für die beiden Roboter "I" und "II". Alle Kanten des Stelle/Transition-Netzes besitzen das Einheitsgewicht "1". Auf eine Beschriftung der Kanten mit ihren Kantengewichten konnte daher in Abb. 32 verzichtet werden. Die Inzidenzmatrix des Stelle/Transition-Netzes folgt in Abb. 33.

In einem ersten Schritt werden Stellen aus dem zugrundeliegenden Stelle/Transition-Netz zu aggregierten Stellen verdichtet. Dadurch werden einerseits die zwei Handhabungsprozesse der beiden Roboter als ähnlich klassifiziert. Daher werden die beiden Teilnetze, die im Stelle/Transition-Netz mit ihren Flüssen von Kopien der Basismarke die Handhabungsprozesse von Roboter "I" und "II" repräsentierten, zu einem Teilnetz zusammengefaltet. Zugleich werden die beiden Stellen s_5 und s_6 (s_7 und s_8) zu einer aggregierten Stelle $s_{5/6}$ ($s_{7/8}$) verdichtet. Um weiterhin zwischen den Handhabungsprozessen der beiden Roboter unterscheiden zu können, müssen eine Markenfarbe "I" für den Handhabungsprozeß des Roboters "I" und eine Markenfarbe "II" für den Handhabungsprozeß des Roboters "II" eingeführt werden. Andererseits werden auch die zwei Werkstückflüsse entlang der beiden Fertigungslinien "A" und "B" als ähnlich eingestuft. Daher werden auch die beiden werkstückflußbeschreibenden Teilnetze des Stelle/Transition-Netzes zu einem Teilnetz zusammengefaltet. Die beiden Stellen s_1 und s_2 (s_3 und s_4) werden zu einer aggregierten Stelle $s_{3/4}$ ($s_{7/8}$) verdichtet. Wiederum werden zwei Markenfarben benutzt, und zwar die Markenfarbe "A" für Bewegungen von Werkstücken auf der Fertigungslinie "A" sowie die Markenfarbe "B" für Bewegungen von Werkstücken auf der Fertigungslinie "B". Es resultiert das kompaktere Höhere Netz, dessen graphische Repräsentation in Abb. 34 wiedergegeben ist. Es umfaßt nur noch zwei Prozesse, die das Verhalten des Miniatur-Produktionssystems beschreiben: Der eine Handhabungsprozeß wird durch den Fluß von Markenkopien der beiden roboterbezogenen Farben "I" und "II" über die Stellen $s_{5/6}$ und $s_{7/8}$ konstituiert. Den einen Werkstückfluß bildet der Fluß von Markenkopien der beiden fertigungslinienbezogenen Farben "A" und "B" über die Stellen $s_{1/2}$ und $s_{3/4}$. Die Verhaltensbeschreibung des Miniatur-Produktionssystems ist aber weiterhin vollständig. Denn jeder der zwei Prozesse aus dem Höheren Netz faßt wegen seiner Aufspaltung durch zwei Markenfarben zwei Prozesse aus dem vorangehenden Niederen Netz zusammen. Insgesamt werden also weiterhin die zwei Handhabungsprozesse für die beiden Roboter "I" und "II" sowie die zwei Werkstückflüsse für die beiden Fertigungslinien "A" und "B" repräsentiert - allerdings in einer kompakteren Netzgraphik mit weniger Stellen und Kanten. Die Anzahl der Transitionen wurde dagegen noch nicht reduziert. Abb. 35 u. 36 zeigen die Inzidenzmatrizen für die Netzgraphik aus Abb. 34. Bei der Inzidenzmatrix aus Abb. 35 wird die Netzgraphik als Repräsentation eines Prädikat/Transition-Netzes aufgefaßt, bei der Inzidenzmatrix aus Abb. 36 als die Repräsentation eines Gefärbten Netzes.

Der zweite Kompaktifizierungsschritt führt zu einer Aggregation von ähnlichen Transitionen. Dabei werden einerseits die zwei Zuordnungstransitionen aus den beiden Fertigungslinien zusammengefaßt. Andererseits werden die zwei Freigabetransitionen aus den beiden Fertigungslinien ebenso aggregiert. Da auf diese Weise keine prozeßrepräsentierenden Teilnetze zusammengefaltet werden, brauchen keine neuen Markenfarben eingeführt zu werden. Statt dessen werden jetzt jeweils zwei Transitionen aggregiert. Jede dieser beiden Transitionen kann in zwei verschiedenen Modi geschaltet werden. Die Schaltmodi richten sich danach, wie die fortgeschalteten Markenkopien eingefärbt sind. Aufgrund der Auswahlmöglichkeit zwischen zwei Schaltmodi für jede Transition ist es nicht mehr möglich, die Ein- und Ausgangskanten der beiden aggregierten Transitionen mit konstanten Markenkopien zu beschriften. Vielmehr handelt es sich um variable Kantenanschriften. Die Variablen $x_{1/2}$ und $x_{3/4}$ vertreten jeweils eine der beiden Fertigungslinien "A" und "B". Sie besitzen daher einen gemeinsamen Definitionsbereich $\{A,B\}$.

Für die korrekte Zuordnung von Robotern zu Werkstücken sorgt die Zuordnungsfunktion "zf". Ihr Nachbereich ist die Potenzmenge pot$\{I,II\}$ aller denkmöglichen Roboterzuordnungen. (Hier gibt ABEL,D. (1990), S. 40f., irrtümlich nur die Robotermenge $\{I,II\}$ als Nachbereich an.) Der Wertebereich der Zuordnungsfunktion "zf" ist die Menge $\{\{I,II\},\{II\}\}$ aller zulässigen Roboterzuordnungen. Die Funktionswerte $zf(x_{1/2})$ und $zf(x_{3/4})$ der beiden Variablen $x_{1/2}$ und $x_{3/4}$ stehen für diejenige Teilmenge der Robotermenge $\{I,II\}$, die bei der Handhabung eines der Werkstücke von Fertigungslinie "A" oder "B" erforderlich ist. Die Abbildungsvorschrift Zuordnungsfunktion der "zf" lautet: $zf(A)=\{I,II\}$ und $zf(B)=\{II\}$. Um die Kohärenz mit der Notation von Kantenanschriften zu wahren, die in dieser Arbeit an der Darstellung von (Multi-)Mengen als formale Summen ausgerichtet ist, wird zusätzlich vereinbart: $<I>+<II> :\Leftrightarrow \{I,II\}$ und $<II> :\Leftrightarrow \{II\}$. Dadurch läßt sich die Abbildungsvorschrift der Zuordnungsfunktionen reformulieren als: $zf(A)=<I>+<II>$ und $zf(B)=<II>$. Diese Abbildungsvorschrift bildet zusammen mit dem Definitionsbereich der Variablen $x_{1/2}$ eine Restriktion für das Schaltverhalten der Transition $t_{1/2}$. Analog dazu wird das Schaltverhalten der Transition $t_{3/4}$ durch eine Restriktion bestimmt, die dem Definitionsbereich der Variablen $x_{3/4}$ und wiederum der Abbildungsvorschrift der Zuordnungsfunktion "zf" besteht.

Abb. 37 stellt die Netzgraphik für das Höhere Netz dar, das aus der voranstehend skizzierten Aggregation von je zwei Transitionen resultiert. Abb. 38 gibt die zugehörige Inzidenzmatrix wieder, wenn das Höhere Netz als ein Prädikat/Transition-Netz aufgefaßt wird. Charakteristisch für diese Matrixvariante ist die Verwendung von Variablen und Funktionswerten als Matrixkoeffizienten. Falls das Höhere Netz aus Abb. 37 ein Gefärbtes Netz darstellen soll, läßt sich die Inzidenzmatrix nicht unmittelbar angeben. Vielmehr müssen - im Gegensatz zum Prädikat/Transition-Netz - zuvor noch die Schaltmodi der beiden Transitionen $t_{1/2}$ und $t_{3/4}$ expliziert werden. Die zwei Schaltmodi der

Transition $t_{1/2}$ werden mit $vb_{1/2.1}$ und $vb_{1/2.2}$ notiert. Sie besitzen die Qualität von Variablenbindungsfunktionen für die Variable $x_{1/2}$. Denn sie sind durch die Abbildungsvorschriften $vb_{1/2.1}$: $x_{1/2} \rightarrow A$ und $vb_{1/2.2}$: $x_{1/2} \rightarrow B$ definiert. Analog werden die zwei Schaltmodi der Transition $t_{3/4}$ mit $vb_{3/4.1}$ und $vb_{3/4.2}$ notiert. Sie stellen Variablenbindungsfunktionen für die Variable $x_{3/4}$ dar. Ihre Abbildungsvorschriften lauten $vb_{3/4.1}$: $x_{3/4} \rightarrow A$ und $vb_{3/4.2}$: $x_{1/2} \rightarrow B$ definiert. Aufgrund des expliziten Differenzierung zwischen den je zwei Schaltmodi der beiden Transitionen tauchen in der Inzidenzmatrix, die das Höhere Netz aus Abb. 37 als Gefärbtes Netz behandelt, weder Variablen noch Funktionswerte auf. Diese Inzidenzmatrix ist in Abb. 39 wiedergegeben.
Auf der dritten Stufe der Netzkompaktifizierung erfolgt eine erneute Aggregation von Stellen. Allerdings findet jetzt - im Gegensatz zur ersten Stellenverdichtung - keine Zusammenfassung von Prozessen statt. Daher ist jetzt mit der Stellenaggregation keine Einführung neuer Markenfarben verknüpft. Die Verdichtung der Stellen beruht auf der Beobachtung, daß die beiden Transitionen $t_{1/2}$ und $t_{3/4}$ einerseits mit den Stellen $s_{1/2}$ und $s_{5/6}$ sowie andererseits mit den Stellen $s_{7/8}$ und $s_{3/4}$ jeweils auf gleichartige Weise verknüpft sind. Daher werden die beiden Stellenpaare zu den aggregierten Stellen $s_{1/2/5/6}$ bzw. $s_{7/8/3/4}$ verdichtet. Jede Transition ist zunächst mit jeder dieser beiden aggregierten Stellen über je zwei Eingangs- und je zwei Ausgangskanten verknüpft. In der graphischen Repräsentation des kompaktifizierten Netzes wird der Umstand genutzt, daß Netze aus graphentheoretischer Perspektive Multigraphen darstellen. Daher lassen sich die Eingangs- und die Ausgangskantenpaare zu jeweils einer (Multi-)Kante verdichten. Sie wird mit der formalen Summe der Gewichte der jeweils zwei zusammengefaßten (Mono-)Kanten gewichtet. Ansonsten ergeben sich keine Veränderungen. Abb. 40 zeigt die Netzgraphik für das Höhere Netz auf der dritten Kompaktifizierungsstufe. In Abb. 41 u. 42 schließen sich die zugehörigen Inzidenzmatrizen an, die das Höhere Netz als Prädikat/Transition-Netz bzw. als Gefärbtes Netz interpretieren.
Der vierte Schritt zu einem noch kompakteren Netz verläuft zunächst analog zur dritten Stufe. Es wird festgestellt, daß die beiden Transitionen $t_{1/2}$ und $t_{3/4}$ mit den Stellen $s_{1/2/5/6}$ bzw. $s_{7/8/3/4}$ auf gleichartige Weise verknüpft sind. Daher lassen sich die beiden Stellen zu nur noch einer aggregierten Stelle $s_{1/2/5/6/7/8/3/4}$ zusammenfassen. Abermals werden je zwei gleichgerichtete Kanten zwischen demselben inzidenten Knotenpaar zu einer Multikante verdichtet und mit einer der formalen Summe der Gewichte der gleichgerichteten Kanten gewichtet. Dabei gilt es allerdings zu berücksichtigen, daß die aggregierte Stelle $s_{1/2/5/6/7/8/3/4}$ nunmehr Markenkopien aufnimmt, die entweder wartende oder aber gehandhabte Werkstücke repräsentieren können. Ebenso kann es sich bei den Markenkopien um die Repräsentation von entweder freien oder aber belegten Robotern handeln. Die Werkstücke von den zwei Fertigungslinien "A" und "B" einerseits sowie die zwei Roboter "I" und "II" andererseits werden durch die bisher eingeführten vier Markenfarben "A", "B", "I" und "II" korrekt differenziert. Für die Unterscheidung zwischen wartenden und gehandhabten Werkstücken sowie zwischen freien und belegten Robotern sorgte bisher die Verschiedenheit der beiden Stellen $s_{1/2/5/6}$ und $s_{7/8/3/4}$. Da diese zwei Stellen zu nur noch einer Stelle aggregiert wurden, entfällt die vorgenannte Unterscheidungsmöglichkeit. Daher wäre es z.B. möglich, daß die Transition $t_{1/2}$ durch ihr Schalten eine Markenkopie mit der Farbe "I", die einen *belegten* Roboter repräsentiert, auf der Stelle $s_{1/2/5/6/7/8/3/4}$ ablegt. Auf diese Markenkopie dürfte ein späteres Schalten derselben Transition $t_{1/2}$ erst dann zugreifen, wenn der Roboter "I" zwischenzeitlich durch ein Schalten der Transition $t_{3/4}$ wieder freigegeben worden wäre. Der Markenkopie mit der Farbe "I" kann auf der Stelle $s_{1/2/5/6/7/8/3/4}$ aber nicht angesehen werden, ob sie entweder einen *freien* oder aber einen *belegten* Roboter repräsentiert. Daher läßt sich nicht ausschließen, daß die Transition $t_{1/2}$ unmittelbar nach ihrem ersten Schalten ein zweites Mal schaltet und dabei die Markenkopie mit der Farbe "I" von der Stelle $s_{1/2/5/6/7/8/3/4}$ wieder abzieht. Dies wäre jedoch sachlich inkorrekt, weil wegen des fehlenden zwischenzeitlichen Schaltakts der Transition $t_{3/4}$ der Roboter "I" noch gar nicht freigegeben ist. Die gleiche Verwechslungsmöglichkeit besteht für die Betriebszustände "frei" und "belegt" des Roboters "II". Auf analoge Weise können die Zustände "wartet" und "gehandhabt" für jedes Werkstück von der Fertigungslinie "A" oder "B" konfundiert werden. Um derart inkorrektes Schaltverhalten auszuschließen, muß jede der vier bisher eingeführten Markenfarben "I", "II", "A" und "B" in zwei Subfarben aufgespalten werden. Die Subfarben differenzieren zunächst zwischen den beiden Teilprozessen der Freigabe und Belegung bei Robotern. Es wird vereinbart, daß die Subfarben "If" ("IIf") und "Ib" ("IIb") den freien bzw. belegten Betriebszustand des Roboters "I" ("II") repräsentieren. In analoger Weise unterscheiden Subfarben zwischen den beiden Teilprozessen des Wartens und des Gehandhabtwerdens bei Werkstücken. Es werden die Subfarben "Aw" ("Bw") und "Ag" ("Bg") verwendet, um auszudrücken, daß ein Werkstück von der Fertigungslinie "A" ("B") wartet bzw. gehandhabt wird. Die Restriktionen und Kantengewichte der Transitionen $t_{1/2}$ und $t_{3/4}$ werden entsprechend angepaßt. Dies schließt die Definition der Abbildungsvorschriften für zwei Hilfsfunktionen "hf" und "hr" ein. Sie stimmen die Subfarben von Werkstücken und derselben Fertigungslinie und von Subfarben desselben Roboters aufeinander ab. Das Ergebnis dieser Modifizierungen gibt die Netzgraphik in Abb. 43 wieder. Die Inzidenzmatrizen, die das Höheren Netz entweder als Prädikat/Transition-Netz oder aber als Gefärbtes Netz behandeln, folgen in Abb. 44 bzw. 45.
Auf der fünften - und letzten - Stufe resultiert ein Höheres Netz, das nur noch aus genau einer Stelle, genau einer Transition und einem Kantenpaar besteht. Damit ist die maximal mögliche Kompaktifizierung erreicht. Eine weitere Aggregation von Stellen oder Transitionen ist unmöglich, weil von jeder Knotenart nur noch genau ein Exemplar existiert. Diese letzte Verdichtung bereitet keine besonderen Schwierigkeiten. Sie geht aus der Zusammenfassung der beiden Transitionen $t_{1/2}$ und $t_{3/4}$ zur aggregierten Transition $t_{1/2/3/4}$ hervor. Die Aggregationsmöglichkeit ist offensichtlich, weil die beiden Transitionen mit einen gemeinsamen Stelle $s_{1/2/5/6/7/8/3/4}$ zwei gleichartige 1-Schleifen bilden.

Die Teilnetze dieser beiden 1-Schleifen werden durch Faltung zu nur noch einer 1-Schleife zusammengefaßt. Die resultierende 1-Schleife fällt mit dem maximal kompaktifizierten Gesamtnetz zusammen. Es ist lediglich zu beachten, daß die je zwei Schaltmodi der beiden Transitionen $t_{1/2}$ und $t_{3/4}$ nicht identisch sind. Daher erhält die aggregierte Transition $t_{1/2/3/4}$ insgesamt vier Schaltmodi. Die Variablenbindungsfunktionen der vier Schaltmodi von Transition $t_{1/2/3/4}$ werden so festgelegt, daß eine eineindeutige Zuordnung mit den zweimal zwei Schaltmodi der beiden zusammengefaßten Transitionen $t_{1/2}$ und $t_{3/4}$ existiert. Abb. 46 präsentiert die Netzgraphik für das maximal verdichtete Höhere Netze. Die zugehörigen Inzidenzmatrizen aus der Perspektive eines Prädikat/Transition-Netzes und eines Gefärbten Netzes folgen in Abb. 47 bzw. 48.

Abschließend wird auf drei charakteristische Eigenschaften der hier erläuterten Netzkompaktifizierung aufmerksam gemacht. Die erste Eigenart betrifft zunächst die Inzidenzmatrizen für Gefärbte Netze. Diese Inzidenzmatrizen enthalten immer genau so viele Zeilenvektoren, wie im jeweils betrachteten Netz verschiedene Markenfarben definiert sind. Da nur endliche Inzidenzmatrizen mathematisch bewältigt werden können, folgt daraus unmittelbar: Gefärbte Netze lassen sich nur anwenden, sofern die Menge aller Markenfarben endlich ist. Dabei handelt es sich um eine spezielle Finitheitsprämisse für Gefärbte Netze, die später näher beleuchtet wird. Dagegen hängen weder die Zeilennoch die Spaltenanzahlen der Inzidenzmatrizen für Prädikat/Transition-Netze von der Anzahl verschiedener Markenfarben ab. Daher sind Prädikat/Transition-Netze nicht auf die Prämisse endlich vieler unterschiedlicher Markenfarben beschränkt.

Ein zweites Charakteristikum läßt sich am leichtesten erkennen, wenn die Inzidenzmatrizen für die gesamte Netzfolge - beginnend mit dem ursprünglichen Stelle/Transition-Netz der Abb. 32 und endend mit dem kompaktesten Höheren Netz der Abb. 46 - untereinander verglichen werden. Dabei zeigt sich eine typische Abweichung zwischen der Inzidenzmatrizen-Entwicklung für Prädikat/Transition-Netze einerseits und für Gefärbte Netze andererseits. Die Inzidenzmatrizen für Prädikat/Transition-Netze werden mit jeder Netzkompaktifizierung um Zeilen- oder Spaltenvektoren verkürzt. Zugleich nimmt in entgegengesetzter Weise die Komplexität der verbleibenden Matrixkoeffizienten zu. Die Inzidenzmatrix des maximal kompaktifizierten Prädikat/Transition-Netzes besitzt schließlich nur noch genau eine Zeile und genau eine Stelle. Entsprechend kompliziert fällt der genau eine Matrixkoeffizient aus. Die Inzidenzmatrizen der Gefärbten Netze entwickeln sich in vollkommen anderer Weise. Die Anzahlen der Zeilen- und Spaltenvektoren der Inzidenzmatrizen bleiben trotz zunehmender Netzkompaktifizierung stets gleich groß. Ebenso verändert sich die Komplexität der Matrixkoeffizienten nicht: Sie stellen immer ganzzahlige Konstanten dar. Diese ganzzahlige Konstanz der Matrixkoeffizienten erleichtert die rechentechnische Anwendung der Inzidenzmatrizen von Gefärbten Netzen erheblich. In einer groben Annäherung kann die Komplexität einer Inzidenzmatrix einerseits durch die Anzahl ihrer Zeilen und Spalten sowie andererseits durch die Kompliziertheit ihrer Koeffizienten beschrieben werden. Bei den Inzidenzmatrizen der Prädikat/Transition-Netze änderten sich die beiden Bestimmungsgrößen der Inzidenzmatrix-Komplexität in entgegengesetzter Weise. Bei den Inzidenzmatrizen der Gefärbten Netze verhielten sich dagegen die beiden Bestimmungsgrößen der Inzidenzmatrix-Komplexität invariant. Daher läßt sich als Groburteil festhalten: Die Komplexität der Inzidenzmatrizen bleibt trotz zunehmender Kompaktheit der Netze sowohl bei Prädikat/Transition-Netzen als auch bei Gefärbten Netzen tendenziell konstant.

Ein dritter bemerkenswerter Gesichtspunkt betrifft den Zusammenhang zwischen allen Netzen aus der Kompaktifizierungsfolge: Jedes kompaktere Netz kann durch "Entfaltung" seiner aggregierten Stellen und Transitionen wieder in jedes beliebige, weniger kompakte Netz zurücktransformiert werden. Dies für jedes Netzpaar aufzeigen, würde hier zu weit führen. Statt dessen wird nur ein verdeutlichendes Beispiel betrachtet. Es betrifft die beiden Antipoden der Kompaktifizierungsfolge, also das maximal kompaktifizierte Höhere Netz aus Abb. 46 und das überhaupt nicht kompaktifizierte Stelle/Transition-Netz aus Abb. 32. Die Vorgehensweise der Entfaltung liegt unmittelbar auf der Hand, wenn das maximal kompaktifizierte Höhere Netz als ein Gefärbtes Netz behandelt wird. Dann steht seine Inzidenzmatrix aus Abb. 48 zur Verfügung. Ein Vergleich mit der Inzidenzmatrix aus Abb. 33 für das Stelle/Transition-Netz zeigt, daß beide Inzidenzmatrizen die gleichen Zeilen- und Spaltenanzahlen besitzen. Darüber hinaus erweisen sich die Matrixkoeffizienten als identisch, wenn zuvor eine geeignete Vertauschung von Zeilen- und Spaltenvektoren vorgenommen worden ist. Es wird darauf verzichtet, die mühseligen Vertauschungen hier im einzelnen zu demonstrieren. Überzeugender erscheint es, die Möglichkeit der entfaltenden Rücktransformation in graphischer Weise zu veranschaulichen. Dabei werden die 8 Marken(sub)farben des Gefärbten Netzes in eineindeutiger Weise den 8 Stellen eines Stelle/Transition-Netzes zugeordnet. Dies entspricht einem zeilenweisen Lesen und Umgruppieren der Inzidenzmatrix des Gefärbten Netzes. Zugleich werden die vier Schaltmodi für die eine Transition des Gefärbten Netzes in eineindeutiger Weise den 4 Transitionen eines Stelle/Transition-Netzes zugewiesen. Das korrespondiert mit einem spaltenweisen Lesen und Umgruppieren der Inzidenzmatrix des Gefärbten Netzes. Schließlich wird jede der 8 verschiedenfarbigen Markenkopien des Gefärbten Netzes durch eine Kopie der einen farblosen Basismarke eines Stelle/Transition-Netzes ersetzt. Die Kopie der Basismarke wird dabei auf derjenigen Stelle abgelegt, die zuvor der Marken(sub)farbe der jeweils substituierten gefärbten Markenkopie eineindeutig zugeordnet wurde. Schließlich resultiert das Stelle/Transition-Netz, das in Abb. 49 wiedergegeben ist. Die zugrundeliegenden Korrespondenzen zwischen Markenfarben und Stellen einerseits sowie Schaltmodi und Transitionen andererseits sind als zusätzliche Stellen- bzw. Transitionenanschriften festgehalten. Ein Vergleich dieses Stelle/Transition-Netzes mit dem ursprünglich vorgegebenen Stelle/Transition-Netz aus Abb. 32 zeigt, daß beide Netze "im wesentlichen" zusammenfallen. Sie unterscheiden sich lediglich hinsichtlich der natürlichsprachlichen

Knotennamen sowie der verdeutlichenden Umrandungen und Schraffuren. Diese drei Aspekte wurden aber weder bei den sukzessiven Netzkompaktifizierungen noch bei der zurücktransformierenden Netzentfaltung berücksichtigt. Daher ist ihr Auseinanderklaffen unerheblich.

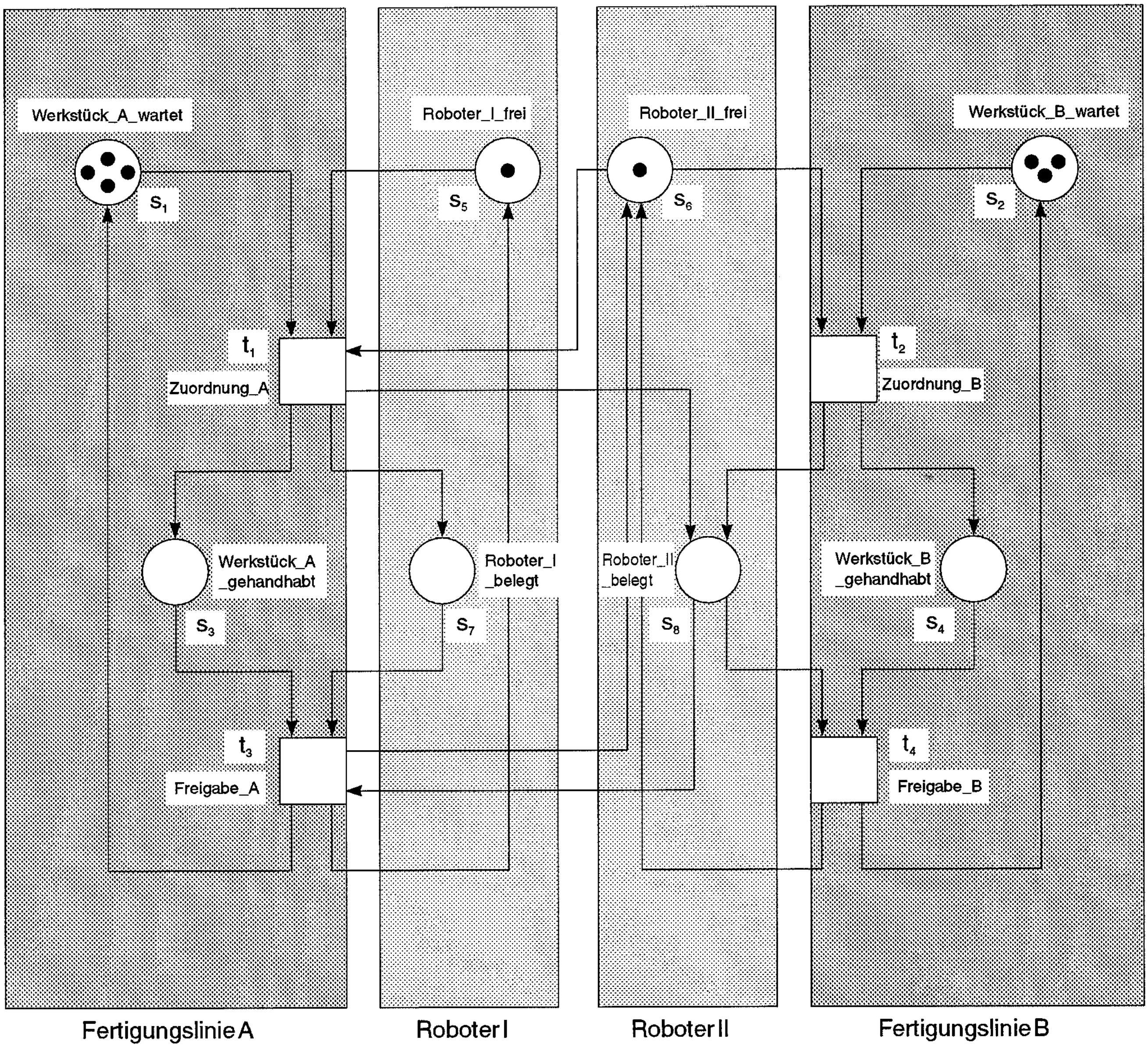

Abb. 32: Stelle/Transition-Netz für ein Miniatur-Produktionssystem

Transitionen / Stellen	t_1	t_2	t_3	t_4
s_1	-1	0	1	0
s_2	0	-1	0	1
s_3	1	0	-1	0
s_4	0	1	0	-1
s_5	-1	0	1	0
s_6	-1	-1	1	1
s_7	1	0	-1	0
s_8	1	1	-1	-1

Abb. 33: Inzidenzmatrix für das Stelle/Transition-Netz aus Abb. 32

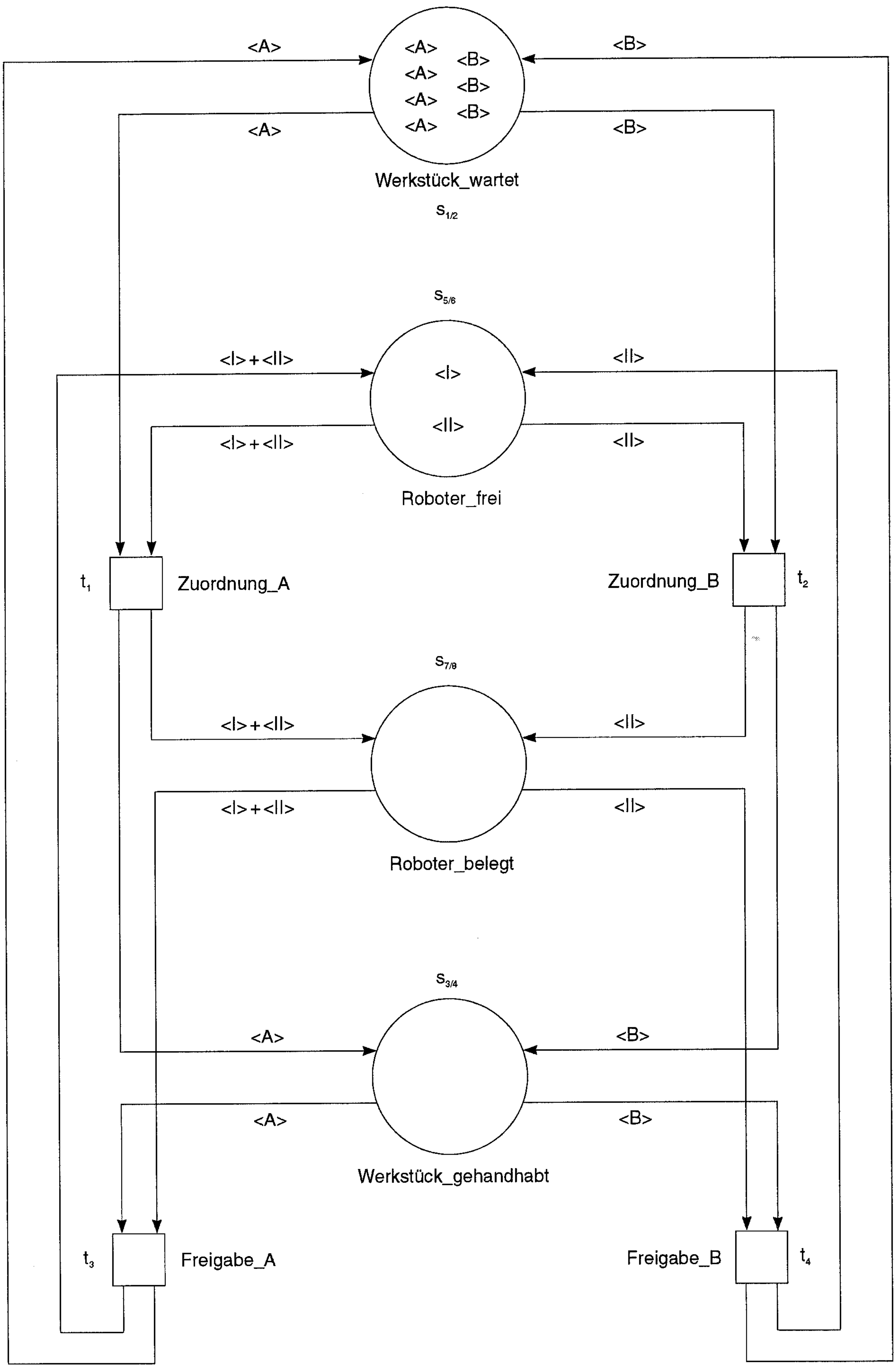

Abb. 34: Höheres Netz für das Miniatur-Produktionssystem aus Abb. 32 mit aggregierten Stellen

Stellen \ Transitionen	t_1	t_2	t_3	t_4
$S_{1/2}$	-A	-B	A	B
$S_{3/4}$	A	B	-A	-B
$S_{5/6}$	-I-II	-II	I+II	II
$S_{7/8}$	I+II	II	-I-II	-II

Abb. 35: Inzidenzmatrix für das Höhere Netz aus Abb. 34, aufgefaßt als ein Prädikat/Transition-Netz

Stellen \ Transitionen		t_1	t_2	t_3	t_4
$S_{1/2}$	A	-1	0	1	0
	B	0	-1	0	1
$S_{3/4}$	A	1	0	-1	0
	B	0	1	0	-1
$S_{5/6}$	I	-1	0	1	0
	II	-1	-1	1	1
$S_{7/8}$	I	1	0	-1	0
	II	1	1	-1	-1

Abb. 36: Inzidenzmatrix für das Höhere Netz aus Abb. 34, aufgefaßt als ein Gefärbtes Netz

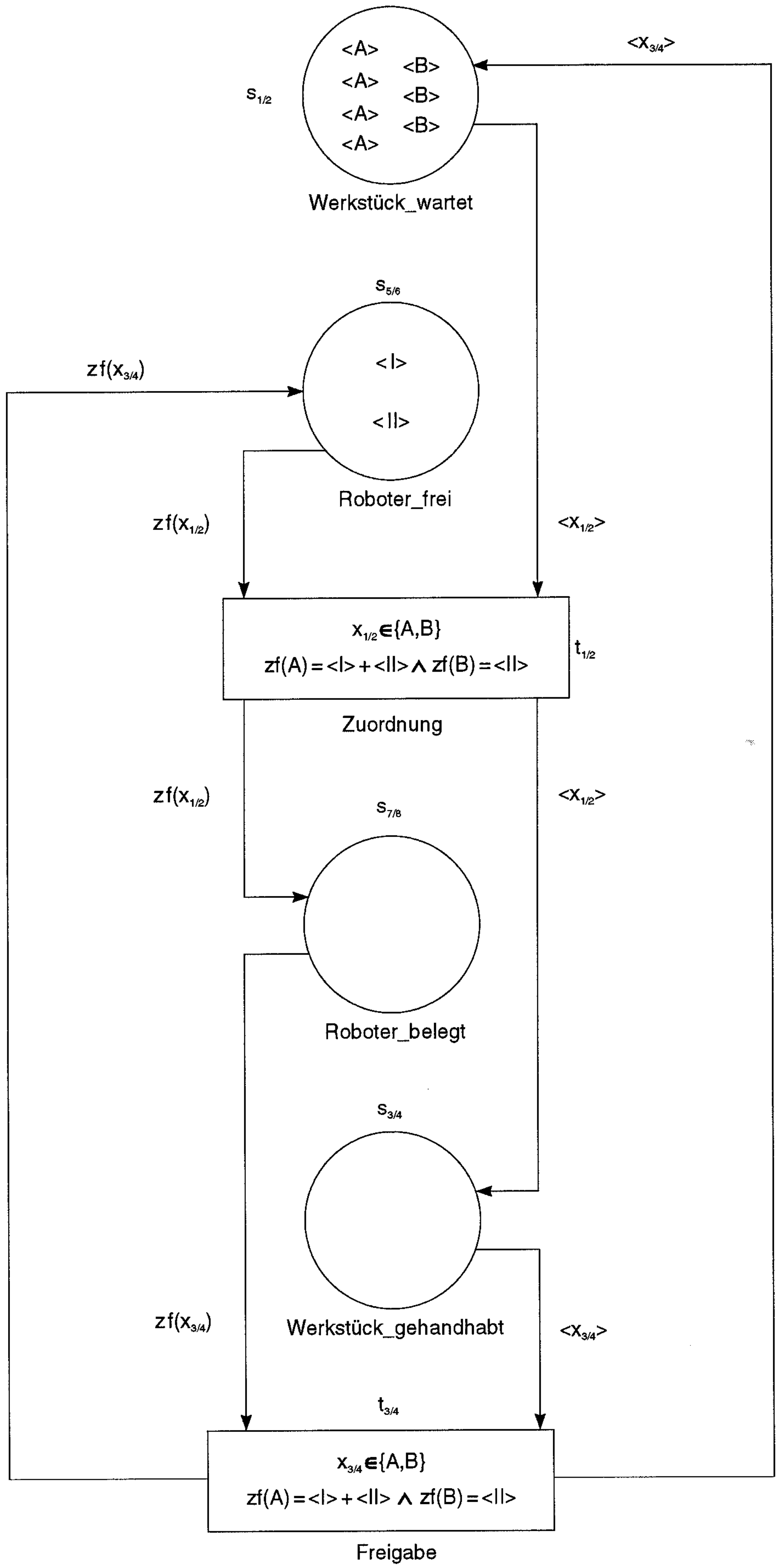

Abb. 37: Höheres Netz für das Miniatur-Produktionssystem aus Abb. 32 mit aggregierten Stellen und aggregierten Transitionen

Transitionen / Stellen	$t_{1/2}$	$t_{3/4}$
$s_{1/2}$	$-x_{1/2}$	$x_{3/4}$
$s_{3/4}$	$x_{1/2}$	$-x_{3/4}$
$s_{5/6}$	$-zf(x_{1/2})$	$zf(x_{3/4})$
$s_{7/8}$	$zf(x_{1/2})$	$-zf(x_{3/4})$

Abb. 38: Inzidenzmatrix für das Höhere Netz aus Abb. 37, aufgefaßt als ein Prädikat/Transition-Netz

Transitionen / Stellen		$t_{1/2}$		$t_{3/4}$	
		$vb_{1/2.1}: x_{1/2} \rightarrow A$	$vb_{1/2.2}: x_{1/2} \rightarrow B$	$vb_{3/4.1}: x_{3/4} \rightarrow A$	$vb_{3/4.2}: x_{3/4} \rightarrow B$
$s_{1/2}$	A	-1	0	1	0
	B	0	-1	0	1
$s_{3/4}$	A	1	0	-1	0
	B	0	1	0	-1
$s_{5/6}$	I	-1	0	1	0
	II	-1	-1	1	1
$s_{7/8}$	I	1	0	-1	0
	II	1	1	-1	-1

Abb. 39: Inzidenzmatrix für das Höhere Netz aus Abb. 37, aufgefaßt als ein Gefärbtes Netz

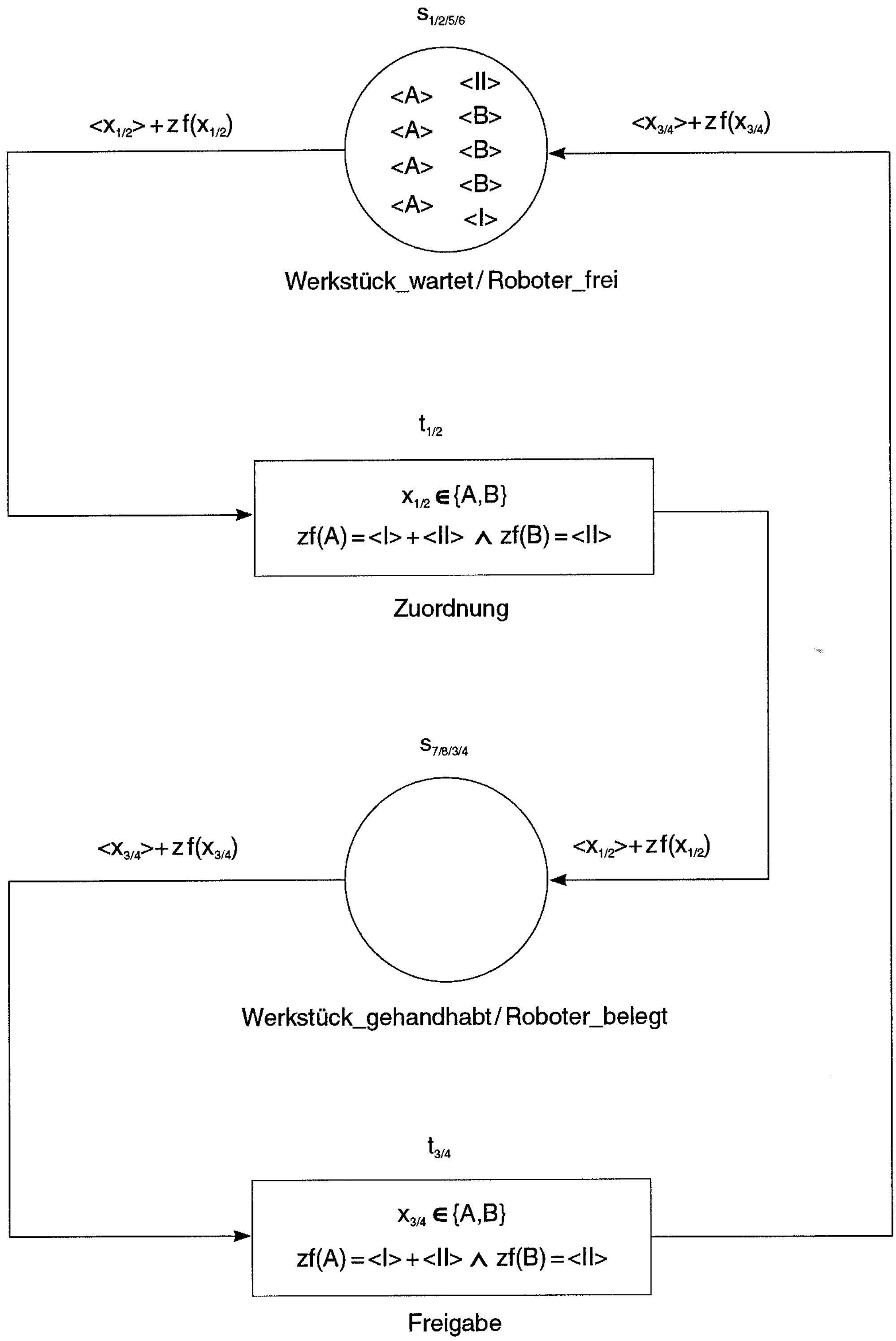

Abb. 40: Höheres Netz für das Miniatur-Produktionssystem aus Abb. 32
mit zweifach aggregierten Stellen und mit aggregierten Transitionen

Transitionen / Stellen	$t_{1/2}$	$t_{3/4}$
$s_{1/2/5/6}$	$-x_{1/2} - zf(x_{1/2})$	$x_{3/4} + zf(x_{3/4})$
$s_{7/8/3/4}$	$x_{1/2} + zf(x_{1/2})$	$-x_{3/4} - zf(x_{3/4})$

Abb. 41: Inzidenzmatrix für das Höhere Netz aus Abb. 40, aufgefaßt als ein Prädikat/Transition-Netz

Transitionen / Stellen		$t_{1/2}$		$t_{3/4}$	
		$vb_{1/2.1} : x_{1/2} \rightarrow A$	$vb_{1/2.2} : x_{1/2} \rightarrow B$	$vb_{3/4.1} : x_{3/4} \rightarrow A$	$vb_{3/4.2} : x_{3/4} \rightarrow B$
$s_{1/2/5/6}$	A	-1	0	1	0
	B	0	-1	0	1
	I	-1	0	1	0
	II	-1	-1	1	1
$s_{7/8/3/4}$	A	1	0	-1	0
	B	0	1	0	-1
	I	1	0	-1	0
	II	1	1	-1	-1

Abb. 42: Inzidenzmatrix für das Höhere Netz aus Abb. 40, aufgefaßt als ein Gefärbtes Netz

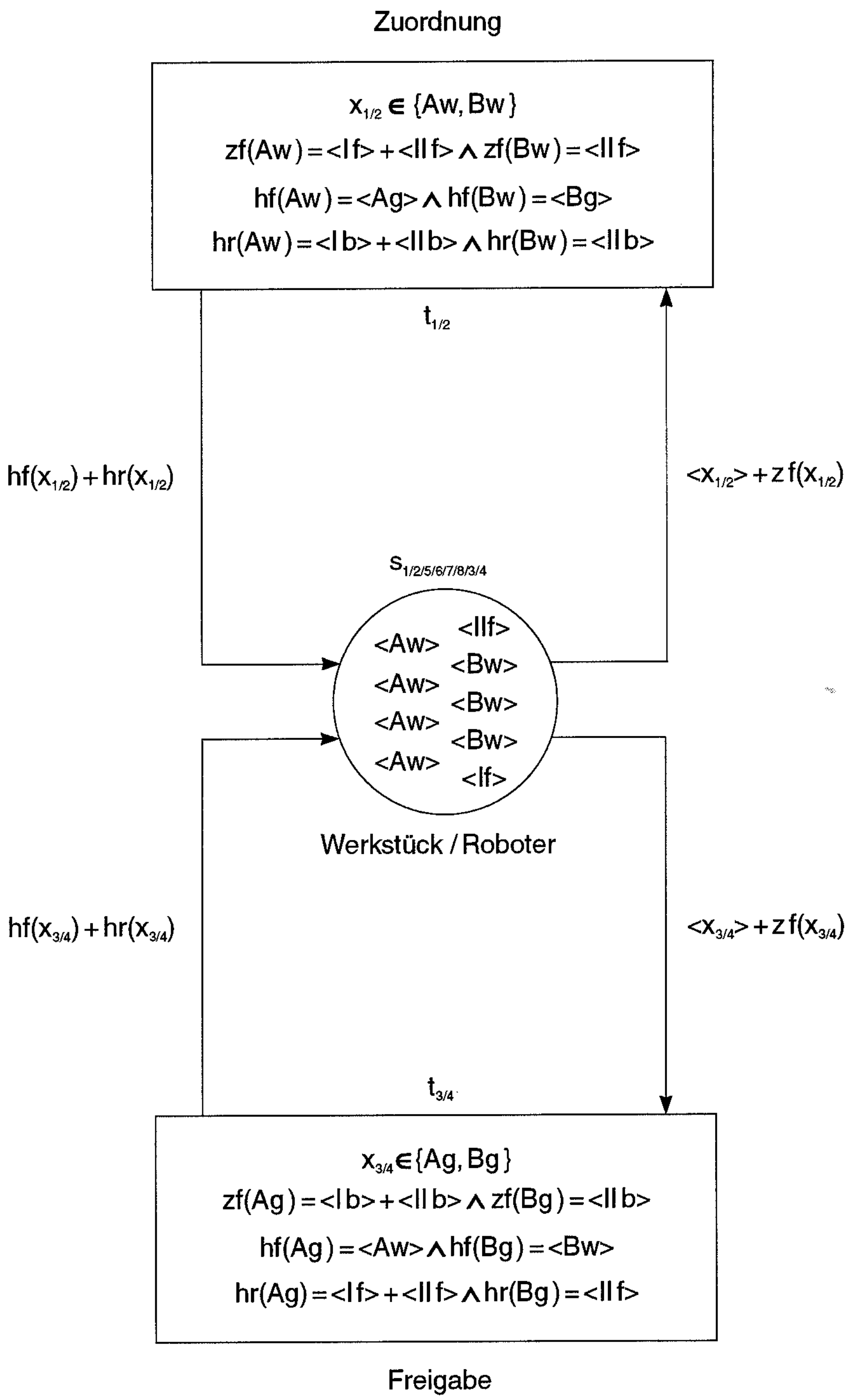

Abb. 43: Höheres Netz für das Miniatur-Produktionssystem aus Abb. 32
mit dreifach aggregierten Stellen und mit aggregierten Transitionen

Transitionen / Stellen	$t_{1/2}$	$t_{3/4}$
$s_{1/2/5/6/7/8/3/4}$	$hf(x_{1/2}) + hr(x_{1/2})$ $- \langle x_{1/2} \rangle - zf(x_{1/2})$	$hf(x_{3/4}) + hr(x_{3/4})$ $- \langle x_{3/4} \rangle - zf(x_{3/4})$

Abb. 44: Inzidenzmatrix für das Höhere Netz aus Abb. 43, aufgefaßt als ein Prädikat/Transition-Netz

Transitionen / Stellen		$t_{1/2}$		$t_{3/4}$	
		$vb_{1/2.1}{}^*: x_{1/2} \rightarrow Aw$	$vb_{1/2.2}{}^*: x_{1/2} \rightarrow Bw$	$vb_{3/4.1}{}^*: x_{3/4} \rightarrow Ag$	$vb_{3/4.2}{}^*: x_{3/4} \rightarrow Bg$
$s_{1/2/5/6/7/8/3/4}$	Aw	-1	0	1	0
	Ag	1	0	-1	0
	Bw	0	-1	0	1
	Bg	0	1	0	-1
	If	-1	0	1	0
	Ib	1	0	-1	0
	IIf	-1	-1	1	1
	IIb	1	1	-1	-1

Abb. 45: Inzidenzmatrix für das Höhere Netz aus Abb. 43, aufgefaßt als ein Gefärbtes Netz

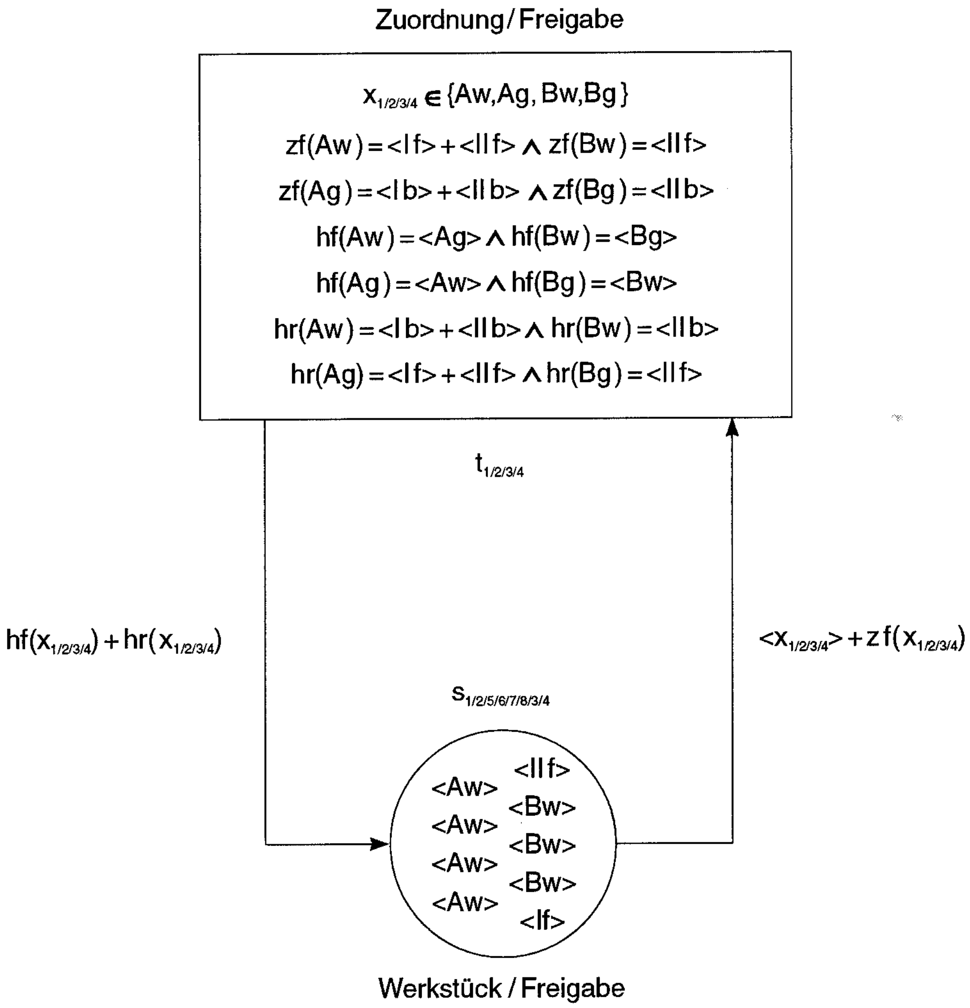

Abb. 46: maximal kompaktifiziertes Höheres Netz für das Miniatur-Produktionssystem aus Abb. 32 mit dreifach aggregierten Stellen und mit zweifach aggregierten Transitionen

Stellen \ Transitionen	$t_{1/2/3/4}$
$s_{1/2/5/6/7/8/3/4}$	$hf(x_{1/2/3/4}) + hr(x_{1/2/3/4}) - <x_{1/2/3/4}> - zf(x_{1/2/3/4})$

Abb. 47: Inzidenzmatrix für das Höhere Netz aus Abb. 46, aufgefaßt als ein Prädikat/Transition-Netz

Stellen \ Transitionen		$t_{1/2/3/4}$			
		$vb_{1/2/3/4.1}$: $x_{1/2/3/4} \rightarrow Aw$	$vb_{1/2/3/4.2}$: $x_{1/2/3/4} \rightarrow Ag$	$vb_{1/2/3/4.3}$: $x_{1/2/3/4} \rightarrow Bw$	$vb_{1/2/3/4.4}$: $x_{1/2/3/4} \rightarrow Bg$
$s_{1/2/5/6/7/8/3/4}$	Aw	-1	1	0	0
	Ag	1	-1	0	0
	Bw	0	0	-1	1
	Bg	0	0	1	-1
	If	-1	1	0	0
	Ib	1	-1	0	0
	IIf	-1	1	-1	1
	IIb	1	-1	1	-1

Abb. 48: Inzidenzmatrix für das Höhere Netz aus Abb. 46, aufgefaßt als ein Gefärbtes Netz

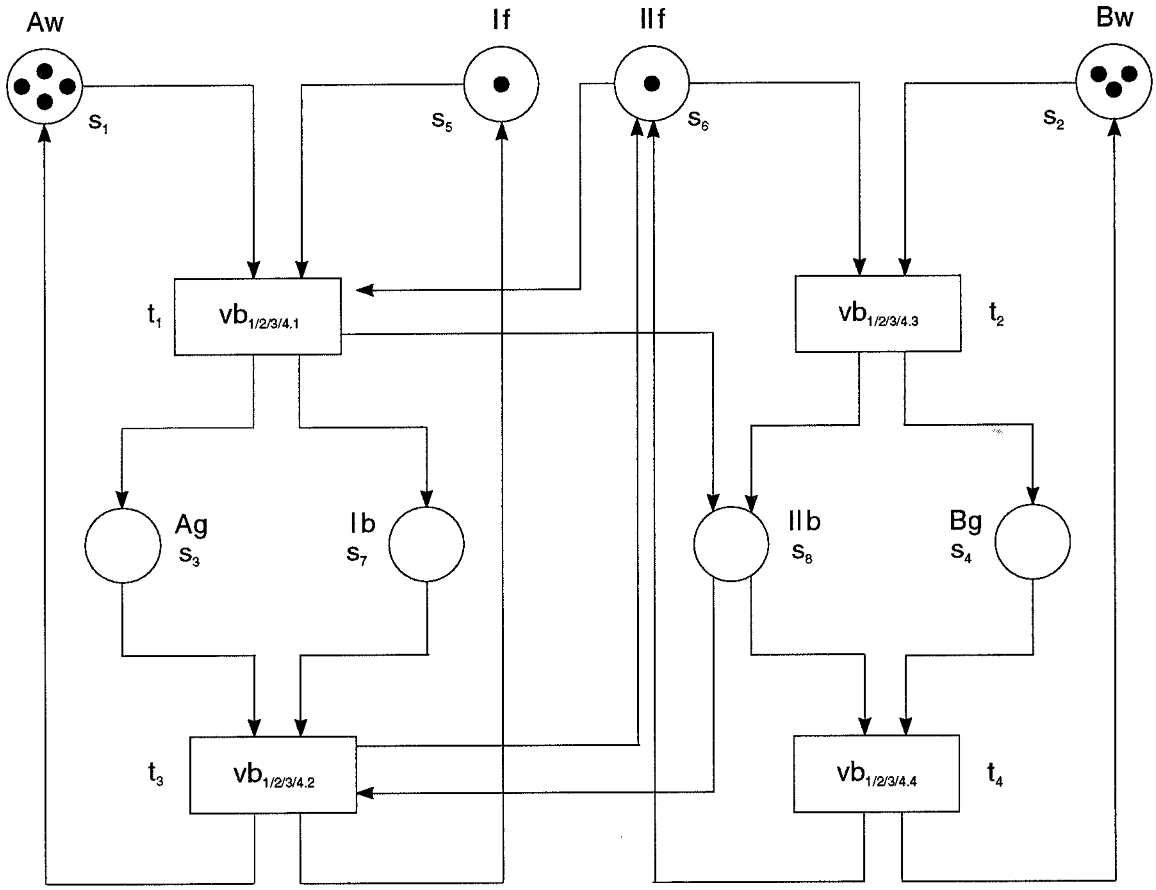

Abb. 49: Stelle/Transition-Netz,

das aus dem maximal kompaktifizierten Höheren Netz der Abb. 46 zurücktransformiert wurde

31) Da jeder Prozeß aus der Verhaltensbeschreibung des Modellierungsobjektes zur Konstitution einer prozeßspezifischen Markenfarbe geführt hat, ist für die Repräsentation *aller* Prozesse eine entsprechend große Anzahl von Markenfarben erforderlich.

32) Vgl. ABEL,D. (1990), S. 42. Vgl. auch die sukzessive Netzkompaktifizierung im voranstehenden Beispiel. Es läßt sich wohl kaum behaupten, daß das letzte, maximal kompaktifizierte Netz der Abb. 46 einen transparenten Eindruck hinterläßt.

33) Darauf wurde schon im Zusammenhang mit Stelle/Transition-Netzen eingegangen. Vgl. auch die spätere Würdigung der Benutzerfreundlichkeit des Petrinetz-Konzepts.

34) Eine solche Erweiterung findet sich in der einschlägigen Petrinetz-Literatur jedoch (noch) nicht. Allerdings kann in diesem Zusammenhang wieder auf das Konzept der objektorientierten Programmierung reflektiert werden, das bereits angesprochen wurde. Denn die Bildung eines hierarchisch organisierten Systems von Markenklassen korrespondiert mit den Objekthierarchien der objektorientierten Programmierung. Das dort übliche "Vererbungsprinzip", durch dessen Vermittlung alle Eigenschaften übergeordneter Objektklassen auf ihnen nachgeordnete Objektsubklassen übergehen, kann analog auf die o.a. Markenklassenhierarchie übertragen werden. Auf diese Weise läßt sich die Menge aller Markenklassen eines Netzes effizient implementieren. Jede übergeordnete Markenklasse "vererbt" ihre Attribute und deren Definitionsbereiche an ihre nachgeordneten Markensubklassen. Auf der untersten Hierarchiestufe liegen alle Markenklassen vollständig definiert vor. Da die Marken einer Markenklasse alle dieselbe, für die Markenklasse spezifische Struktur aus Attributen und Definitionsbereichen besitzen, konstituiert die Hierarchie der Markenklassen zugleich eine entsprechende Hierarchie der zugehörigen Attribute.

35) Eine beschränkte quantitative System-Modellierung lassen bereits die farblosen Basismarken in Stelle/Transition-Netzen zu, da ihre Anzahl unter der jeweils aktuellen Markierung - ggf. in einem ausgezeichneten Teilnetz - gemessen werden kann. Die Trägermenge $\mathcal{N}_0$ der Markierungsfunktion ermöglicht aber nur ganzzahlige Messungen. Dies reicht für die gewöhnlich reellzahlige Messung betriebswirtschaftlicher Sachverhalte jedoch nicht aus. Darüber hinaus wäre es überaus umständlich, die ganzzahligen Ausprägungen eines quantitativen Attributs jeweils durch eine entsprechend große Markenanzahl modellieren zu müssen. Statt dessen kann bei der Verwendung von Attributmarken dieselbe Ausprägung als die Eigenschaft einer einzigen Marke repräsentiert werden.

36) Numerische Attributausprägungen werden fortan auch als Parameterwerte bezeichnet.

37) Hierdurch bleibt die Markenverteilung - mit Ausnahme der neu hinzugefügten oder der aus dem Netz entfernten Marken - konstant. Vereinfachend wird dieser Sachverhalt fortan als "unveränderte Markenverteilung" angesprochen.

38) Damit werden - ebenso mögliche - Bewegungen, die zur ursprünglichen Markenverteilung (und -anzahl) zurückführen, von der weiteren Betrachtung ausgeschlossen. Dies ist plausibel, solange nur die strukturlosen Marken von Stelle/Transition-Netzen zugrundeliegen. Denn bei konstanter Markenverteilung (und -anzahl) wäre im Netz überhaupt keine Änderung erfolgt, so daß eine Schaltwirkung nicht festgestellt werden könnte. "Wirkungslose Schaltwirkungen" stellen aber eine semantische contradictio in se dar, die grundsätzlich nicht weiter berücksichtigt wird.

39) Diese Kombination ist sogar der Normalfall. Zumeist werden Anzahl und Verteilung der Marken durch das Schalten einer Transition simultan verändert. Nur ausgezeichnete Netzklassen - wie z.B. die Netze vom Typ der Zustandsautomaten - beruhen auf der reinen Änderung der Markenverteilung bei konstanter Markenanzahl. Auf den umgekehrten Fall einer reinen Veränderung der Markenanzahl bei konstanter Markenverteilung wird unten näher eingegangen.

40) Auch diese Variante kann mit den beiden vorgenannten kombiniert werden.

41) Vgl. zur nachfolgenden Differenzierung zwischen Objektklassen und deren Instantiierungen durch einzelne Objekte z.B. VON ZIMMERMANN (1990), S. 242.

42) Vgl. zu solchen konzeptuellen Schemata für Datenbanksysteme WINTER,RO. (1991), S. 214ff., 242ff. u. 296ff., insbesondere S. 245ff. (mit speziellem Bezug auf Datenbanken für PPS-Systeme); KLEINER,F. (1991), S. 61ff. (für Datenbanken, die auf Kostenrechnungen für Flexible Fertigungssysteme zugeschnitten sind).

43) Ein Entity ist ein bereichsspezifischer Begriff für die Konzeptualisierung von Datenstrukturen. Er entspricht dem ontologischen Begriff der "Entität" und dem epistemologischen Begriff des Erkenntnisobjekts. Aus der informationstechnischen Perspektive der Gestaltung von Datenbankstrukturen läßt sich ein Entity als kleinste selbständig identifizierbare Datenstruktur auffassen; vgl. CHEN,P.P. (1976), S. 10; NEUMANN,T. (1983), S. 38. Aus dem betriebswirtschaftlichen Blickwinkel des Aufbaus von Unternehmungs-Datenmodellen versteht SCHEER (1988a), S. 20, Entities als "Dinge, die für eine Unternehmung von Interesse sind". Sowohl Identifizierbarkeit als auch Interessenbezug verweisen auf den o.a. epistemologischen Aspekt des Erkenntnisobjekts. Besonders deutlich wird er bei MAYR,H. (1987), S. 499, wo von Entities als "Objekten unserer Anschauung und unseres Denkens" gesprochen wird. Vgl. zu ähnlichen inhaltlichen Füllungen des Entity-Begriffs SINZIG (1983), S. 88f.; LINDGREEN (1987), S.

119ff. u. 126ff., insbesondere S. 127; WAND (1989), S. 539(ff., insbesondere seine Ausdehnung zum konzeptuellen Schema für Unternehmungs-Datenmodellen auf S. 542); WINTER,RO. (1991), S. 217.
Ein Entity wird in dieser Arbeit mit dem weit gefaßten Objektbegriff aus der objektorientierten Systemstrukturierung identifiziert. Wenn ein engerer Objektbegriff vertreten wird, der sich an der umgangssprachlichen Gegenstandsvorstellung ausrichtet, kann ein Entity auch als Oberbegriff zu Objekten i.e.S. und Ereignissen verstanden werden; vgl. NEUMANN,T. (1983), S. 38.

44) Vgl. zum relationalen Architekturkonzept für Datenbanksysteme CODD (1970), S. 377ff.; CODD (1971a), S. 2ff.; CODD (1972), S. 65ff.; WEDEKIND (1979a), S. 367ff.; WEDEKIND (1980), S. 663ff.; WEDEKIND (1981), S. 180ff.; SCHLAGETER (1983), S. 80ff.; APPELRATH (1983), S. 16ff.; SINZIG (1983), S. 78ff.; SCHEER (1987a), S. 193ff.; SCHEER (1988a), S. 29ff.; SCHEER (1989c), S. 26ff.; SCHEER (1991d), S. 154ff.; KLEINER,F. (1991), S. 63ff.

45) Hierauf wurde schon anläßlich der Erörterung identifizierender Markenattribute hingewiesen; vgl. DITTRICH,K. (1989), S. 215.

46) Auf dieses Unternehmensdatenmodell wurde schon einleitend im Zusammenhang mit der datenbezogenen Systemstrukturierung hingewiesen.

47) Näheres zum Entity-Relationship-Konzept bei CHEN,P.P. (1976), S. 10ff.; NEUMANN,T. (1983), S. 37ff.; SINZIG (1983), S. 94ff.; DITTRICH,K. (1985), S. 117f.; HOHENSTEIN (1986), S. 186 u. 188f.; MAYR,H. (1987), S. 498ff.; SCHEER (1988a), S. 20ff.; SCHEER (1989c), S. 17ff.; SCHEER (1991d), S. 19ff., 51ff. u. 95ff.

48) Das Entity-Relationship-Konzept stellt strenggenommen keine Variante relationaler Datenbankarchitekturen dar. Solche Architekturen beschreiben schematisch die Datenstrukturen von konkret implementierten oder implementierbaren Datenbanksystemen. Das Entity-Relationship-Konzept geht dieser Implementierungsebene dagegen logisch voran, indem es einen Strukturierungsansatz für konzeptuelle Datenschemata zur Verfügung stellt. In diesem Sinn hat schon CHEN,P.P. (1976), S. 9f. u. 25ff., sein Konzept als ein abstraktes "Überschema" betrachtet, aus dem die implementierungsbezogenen Datenbankschemata abgeleitet werden können. Zu jenen Architektur-Schemata gehört u.a. auch das relationale Datenbankschema von CODD. Darüber hinaus führen die meisten Implementierungen von konzeptuellen Datenschemata zu realen Datenbanksystemen, deren Architekturen das CODD'sche relationale Datenbankschema erfüllen. Vgl. zu dieser engen Affinität zwischen Entity-Relationship-Konzept einerseits und relationaler Datenbankarchitektur andererseits SCHEER (1988a), S. 29. Vgl. ebenso zum "Überschema"-Charakter des Entity-Relationship-Konzepts seine Anwendung "zur Datenmodellierung ... auf der Metaebene" bei SCHEER (1991d), S. 51.
Schließlich zeigt eine nähere Analyse des Entity-Relationship-Konzepts, daß seine logische Struktur die Qualität einer (sortierten) Prädikatenlogik 1. Stufe besitzt. Vgl. zur prädikatenlogischen Behandlung des Entity-Relationship-Konzepts z.B. NAKANO,R. (1983), S. 552ff., und - weniger konzise - TABOURIER (1983b), S. 571ff. Dabei kommt den Entitytypen oder Entitysets die prädikatenlogische Qualität von (sortierten) Individuenmengen zu. Bei den Individuen kann es sich sowohl um selbständig existierende Objekte als auch um akzidentielle Objekteigenschaften handeln. Die Beziehungstypen oder Relationshipsets entsprechen dagegen mehrstelligen Prädikaten mit flachen Prädikatsargumenten. Das Argument eines gültigen Prädikatsvorkommnisses stellt ein Tupel aus prädikatserfüllenden Individuen (Entities) dar. Ein solches Tupel tritt innerhalb des Entity-Relationship-Konzepts als eine einzelne Beziehung oder Relationship auf. Die Extension eines Prädikats ist die Menge aller prädikatserfüllenden Individuen-Tupel.
Varianten der Prädikatenlogik 1. Stufe, die in der voranstehend skizzierten Weise ausgestaltet sind, werden des öfteren als Relationenlogik bezeichnet. Vgl. dazu die Erläuterungen zur relationalen Ausdruckskraft der Prädikatenlogik. Jedes Konzept, das eine solche relationenlogische Charakteristik aufweist, wird in dieser Arbeit als ein relationaler Ansatz bezeichnet. In diesem Sinne gehören sowohl CHEN's Entity-Relationship-Konzept als auch CODD's relationale Datenbankarchitekturen zur Klasse der relationalen Datenstrukturierungen.

49) Als konventionell werden hier die "Standardapplikationen" des relationalen Ansatzes im betriebswirtschaftlich-administrativen Bereich aufgefaßt. Sie entsprechen dem originären relationalen Ansatz. Aufgrund der nachfolgend partiell angesprochenen Einschränkungen des konventionellen relationalen Ansatzes wurden auch "Nonstandardapplikationen" - vornehmlich für den technischen Bereich von CAD/CAM-Systemen - entwickelt. Auf ihre fortentwickelten relationalen Datenschemata treffen die hier vorgetragenen Anmerkungen nicht mehr zu. Vgl. zu solchen Fortentwicklungen z.B. NEUMANN,T. (1983), S. 42ff.; DITTRICH,K. (1985), S. 116ff., insbesondere S. 117f.; HOHENSTEIN (1986), S. 186ff., insbesondere S. 190ff.; HOHENSTEIN (1987), S. 59ff., insbesondere S. 66ff.; COMMISSION OF THE EUROPEAN COMMUNITIES (1989), S. I-15.
Vgl. allgemein zur Kritik an den beschränkten Ausdrucksmöglichkeiten des Entity-Relationship-Konzepts NEUMANN,T. (1983), S. 40 u. 43; TABOURIER (1983b), S. 565f. (hinsichtlich der Unmöglichkeit, einzelne Individuen oder Beziehungen anstelle von Entitytypen bzw. Beziehungstypen zu repräsentieren); HOHENSTEIN (1986), S. 190 (mittelbar); HOHENSTEIN (1987), S. 59f., 64 u. 66 (mittelbar); COMMISSION OF THE EUROPEAN COMMUNITIES (1989), S. I-15; VON ZIMMERMANN (1990), S. 259; ZELEWSKI (1990c), S. 79 (in bezug auf Zeitgrößen); WINTER,RO. (1991),

S. 221ff., 229f. u. 334f. (der Autor bezieht sich zwar auf relationale Datenbankschemata; aber seine Argumente gelten für das Entity-Relationship-Konzept im Prinzip ebenso).

50) Die flachen Objekte relationaler Ansätze stellen lineare Listen - d.h. Attributtupel - dar. Komplexere Objekte, die hierarchischen Listenstrukturen entsprechen, können nur als relationale Verknüpfungen der flachen Attributtupel-Objekte dargestellt werden. Diese Verknüpfungskomplexe lassen sich aber nicht mehr unmittelbar als Individuen identifizieren. Daher wird eine "natürliche" Repräsentation komplexer realer Objekte mit hierarchisch verschachtelten inneren Strukturen von relationalen Ansätzen verfehlt. Vgl. zu dieser Beschränktheit relationaler Gestaltungskonzepte angesichts komplex strukturierter realer Objekte DITTRICH,K. (1989), S. 215f. Vgl. ebenso zur Notwendigkeit, das Entity-Relationship-Konzept für die Erfassung komplexer Objektstrukturen zu erweitern, HOHENSTEIN (1986), S. 191f.; HOHENSTEIN (1987), S. 59f., 64, 66 u. 68f.

51) Vgl. NEUMANN,T. (1983), S. 42f.; DITTRICH,K. (1989), S. 215.

52) Vgl. NEUMANN,T. (1983), S. 43; DITTRICH,K. (1989), S. 215.

53) Vgl. DITTRICH,K. (1989), S. 215.

54) Vgl. JACKSON,M.A. (1979), S. 29ff. Dabei kennzeichnen "*" eine K_*-fache Iteration mit $K_* \in \mathcal{N}_0$ und "o" eine K_o-fache Selektion (Alternative) mit $K_o \in \mathcal{N}_+$ und $K_o \geq 2$.

5.1.1.2 Abgrenzung von Prädikat/Transition-Netzen

Die bedeutsame Ausweitung der Modellierungsfähigkeit von Petrinetzen, die durch die Verwendung strukturierter Marken eröffnet wird, kommt allen Höheren Netzen zu. Höhere Netze werden bislang vornehmlich in zwei Varianten verwendet: als Prädikat/Transition-Netze[1)] und als Gefärbte Netze[2)]. Zwar wurden beide Netzklassen zunächst unabhängig voneinander konzipiert und in verschiedenartigen Frühformen präsentiert. Aber mittlerweile haben sich Prädikat/Transition-Netze und Gefärbte Netze durch Fortentwicklungen so weit aneinander angenähert, daß sie sich kaum noch unterscheiden[3)]. Daher reicht es im folgenden aus, nur noch auf Prädikat/Transition-Netze als "typische" Höhere Netze einzugehen[4)].

Die Synthetischen Netze, die in den anschließenden Kapitel entfaltet werden, weichen in zwei wesentlichen[5)] Aspekten von Prädikat/Transition-Netzen ab. Der erste Gesichtspunkt betrifft die Interpretation der materiellen Bedeutung - des "ontologischen Status"[6)] - von strukturierten Marken[7)]. Eine wohldefinierte Markenontologie ist zwar für Synthetische Netze konzipiert, nicht aber für Prädikat/Transition-Netze. Der zweite Aspekt erstreckt sich auf die Freizügigkeit bei der Strukturierung formaler Objekte. Die Objektstrukturierung ist bei Prädikat/Transition-Netzen keinen Begrenzungen unterworfen. In der Markenontologie Synthetischer Netze wird sie dagegen in charakteristischer Weise durch strukturelle Integritätsbedingungen eingeschränkt. Zunächst wird auf den ersten Aspekt der Markenontologie eingegangen.

In Synthetischen Netzen werden Attributmarken - und die daraus zusammengesetzten Kompositmarken[8)] - immer als Repräsentanten von entsprechend strukturierten Objekten des jeweils modellierten Realproblems aufgefaßt[9)]. Marken und repräsentierte Objekte sind so definiert, daß sie im Kontext der zugrundeliegenden Problemmodellierung die kleinsten[10)] Einheiten darstellen, die Veränderungen im Netzmodell bzw. im modellierten Realitätsausschnitt beeinflussen und zugleich von diesen betroffen werden können[11)]. Die Bestandteile von Marken- bzw. Objektstruktur nehmen an solchen Veränderungen zwar teil, aber nicht als selbständige Einheiten, sondern nur als Attribute[12)] der jeweils betroffenen Marken bzw. Objekte[13)]. Marken und Objekte stellen somit die kleinsten selbständig existierenden Entitäten in der jeweils vorliegenden Problemmodellierung dar. Marken besitzen als formale Objekte also eine eindeutige ontologische Interpretation durch die jeweils modellierten realen Objekte.

In Prädikat/Transition-Netzen werden dagegen Marken zunächst überhaupt nicht definiert[14)]. Dort wird vielmehr von unstrukturierten "Individuen"[15)] ausgegangen[16)], die grundsätzlich keine Marken darstellen. Diese "Individuen" werden als kleinste Einheiten von Netzveränderungen behandelt[17)]. Aus diesen "Individuen" werden K-Tupel (mit $K \in \mathcal{N}_0$[18)]) als Komplexeinheiten gebildet. Formal entsprechen die "Individuen" den o.a. Markenattributen und die Komplexeinheiten den vorgenannten Attributmarken. Diese Komplexeinheiten werden in manchen Beiträgen zu Prädikat/Transition-Netzen explizit als Marken behandelt[19)], während in anderen der Markenbegriff konsequent durch den des (formalen) K-Tupels ersetzt wird[20)]. Im letzten Fall lassen sich zwar beliebige Attributzusammenstellungen durch freie Kombination aller definierten "Individuen" bilden. Aber Marken als formale Abbilder realer Objekte sind dann im strengen Sinne überhaupt nicht mehr explizit definiert.

Selbst wenn die K-Tupel aus "Individuen" explizit als Marken behandelt werden, so lassen sich diese Komplexeinheiten doch oftmals materiell nicht mehr als Repräsentanten von Objekten eines modellierten Realproblems interpretieren. Denn die Syntax von Prädikat/Transition-Netzen gestattet es, durch das Schalten einer Transition auf deren Ausgangsstellen K-Tupel mit einer vollkommen anderen Struktur abzulegen, als sie den K-Tupeln zukommt, die von ihren Eingangsstellen abgezogen werden. Nur die "Individuen" als Tupelbestandteile werden vom Schalten einer Transition als solche "Individuen" nicht verändert. Würden nun z.B. die K-Tupel auf den Eingangsstellen als Objekte mit den "Individuen" als Objektattributen gedeutet, so könnte das Schalten einer Transition zu neuartigen, artifiziellen "Objekten" führen, die aus den vorgenannten durch willkürliche Kombination ihrer Attribute hervorgegangen sind. Es wäre nicht sichergestellt, daß der Attributmix solcher "Objekte" irgendeinen Bezug zum modellierten Realproblem besäße. Somit ließen sich die K-Tupel auf den Ausgangsstellen nicht mehr notwendig als Repräsentanten von Objekten dieses Problems interpretieren[21)].

Die vorgenannten Schwierigkeiten resultieren aus dem Umstand, daß in Prädikat/Transition-Netzen der ontologische Status der K-Tupel nicht eindeutig geklärt ist. Es bleibt offen, ob diese Individuen-Tupel selbständig existenzfähige Objekte darstellen oder nur lockere Aggregate von Individuen ohne eigenständige Existenz. Somit können die K-Tupel einerseits durchaus reale Objekte repräsentieren. Andererseits ist es aber ebenso möglich, daß ein K-Tupel nur einzelne Attribute eines realen Objekts herausgreift oder Attribute aus verschiedenen realen Objekte auf derselben Modellierungsebene vermengt. In dieser Hinsicht weisen die K-Tupel von Prädikat/ Transition-Netzen dieselbe ontologische Indifferenz wie die Datenobjekte relationaler Datenschemata und wie die Objekte bei der objektorientierten Systemgestaltung auf[22)]. Auch beim relationalen bzw. objektorientiertem Konzept wird zwischen Attributen als Objekten ohne eigenständige Existenzfähigkeit einerseits und selbständig existierenden Objekten andererseits formal nicht differenziert. Beide ontologisch verschiedenen Objektkategorien werden durch die gleichen Daten- bzw. Objekttypen erfaßt.

Darüber hinaus erweist sich der Begriff des "Individuums" von Prädikat/Transition-Netzen als höchst dubios, falls er im Sinne eines individuellen Objekts gedeutet wird. Denn von solchen Individuen können unter derselben Netzmarkierung beliebig viele Kopien existieren. Dies widerspricht jedoch dem intuitiven Begriffsverständnis von der Einmaligkeit jedes Individuums[23)]. Ferner wird mitunter der Individuenbegriff auch auf (individuelle) Marken bezogen. Dies führt zu erheblichen Begriffsverwirrungen, weil derselbe Begriff "Individuum" sowohl auf die o.a. "Individuen" als Markenbestandteilen als auch auf Marken als K-Tupel aus solchen "Individuen" bezogen wird[24)].

Schließlich können Prädikat/Transition-Netze nur solche beweglichen Objekte unmittelbar repräsentieren, die bereits im Kontext identifizierender Markenattribute als "flache" Objekte eingeführt wurden. Die K-Tupel aus Individuen stellen lineare Listen aus objektbeschreibenden Attributen dar. Objekte mit einer komplexeren, hierarchisch verschachtelten inneren Struktur können auf diese Weise nicht direkt modelliert werden[25)]. Insofern implizieren Prädikat/Transition-Netze eine "flache" oder "tupuläre" Ontologie für die darstellbaren beweglichen Objekte. Die hier entfalteten Synthetischen Netzen erlauben hingegen, eine "tiefe" oder "hierarchische" Marken- und Objektontologie zu entwerfen. Durch ihre zusammengesetzten strukturierten Marken ist es möglich, Objekte mit hierarchisch verschachtelten Attributstrukturen[26)] zu repräsentieren. Ebenso lassen sich rekursive Objektkompositionen[27)] beliebiger Komplexität bilden.

Das Markierungskonzept von Prädikat/Transition-Netzen leidet also unter den Schwierigkeiten einer mitunter fehlenden expliziten Markendefinition, eines nicht immer sichergestellten Bezugs zu Objekten des modellierten Realproblems, einer Gefahr von ungewollten Fernwirkungen bei der Modellgestaltung, eines problematischen Gebrauchs des Individuenbegriffs und einer bloß flachen Objektontologie. Um diese Modellierungsprobleme zu vermeiden, wird das Konzept strukturierter Marken im folgenden Kapitel abweichend von der üblichen Gestaltung der Prädikat/Transition-Netze formuliert. Dennoch wird es so ausgestaltet, daß der formale Kalkül von Prädikat/Transition-Netze weiterhin angewendet werden kann. Der wesentliche Unterschied besteht darin, daß Marken nicht als abgeleitete Zusammenfassungen anderer Entitäten - der "Individuen" - definiert werden. Statt dessen werden in Synthetischen Netzen die Marken explizit als ontologische Basiskonstrukte eingeführt, die nicht mehr in andere, selbständig existenzfähige Konstrukte zerlegt werden können[28)].

Der zweite wesentliche Unterschied zwischen Prädikat/Transition-Netzen und Synthetischen Netzen betrifft die Möglichkeit, komplexe formale Objekte aus einfacheren formalen Objekten zusammenzusetzen. Prädikat/Transition-Netze erlauben es, Operationen auf formale Objekte anzuwenden, ohne hierbei die strukturelle Integrität der Operationsobjekte beachten zu müssen[29]. Unter der strukturellen Objektintegrität wird die Aufrechterhaltung (Persistenz[30]) der internen Strukturierung formaler Objekte verstanden, solange kein hinreichender und explizit angegebener Grund vorliegt, die Objektstruktur zu verändern. Eine solche Strukturpersistenz ist der Normalfall für die formalsprachliche Modellierung realer Objekte. Denn in den modellierten Realitätsausschnitten führen reale Aktionen im allgemeinen nur zu Veränderungen der Ausprägungen von Objekteigenschaften im weitesten Sinne[31]. Die Objektstruktur selbst, die durch Anzahl, Art und Anordnung der Objekteigenschaften konzeptualisiert wurde, wird dagegen durch die Aktionsausführungen in der Regel nicht modifiziert[32]. Von dieser Strukturkonstanz der Konzeptualisierung realer Objekte wird fortan als Persistenzprämisse ausgegangen[33].

Falls eine Aktion auf reale Objekte angewendet wird und diese Objekte als solche bei der Aktionsausführung erhalten bleiben[34], so müssen aufgrund der Persistenzprämisse auch die Objektstrukturen unverändert fortbestehen. Dies bedeutet für ein Modell, in dem die reale Aktion auf eine formale Operation und die realen Objekte auf formale Objekte abgebildet werden, eine entsprechende Integritätsbedingung für die Erhaltung der formalen Objektstrukturen: Solange ein reales Objekt strukturell unverändert bleibt, dürfen auch die formalen Objekte, die dieses Realobjekt in einem formalsprachlichen Modell repräsentieren, strukturell nicht verändert werden (strukturelle Integritätsbedingung). Eine Integritätsverletzung tritt dagegen dann ein, wenn die formalen Objekte, die aus der Operationsausführung als Operationsoutput hervorgehen, strukturell von denjenigen formalen Objekten des Operationsinputs abweichen, auf welche die Operation angewendet worden ist. Eine Integritätsverletzung liegt ebenso vor, falls die formalen Objekte bei den Operationsausführungen entweder vernichtet oder aber erschaffen werden, obwohl die jeweils repräsentierten Realobjekte fortwährend Bestand haben[35]. Denn sowohl die Strukturauslöschung einer formalen Objektvernichtung als auch die Strukturemergenz einer formalen Objekterschaffung widersprechen der strukturellen Persistenz der zugrundeliegenden Realobjekte.

Strukturelle Integritätsverletzungen erfolgen in manchen Netzmodellen vor allem[36] dadurch, daß Informationen über ein modelliertes reales Objekt, die nach einer Operationsausführung in einem lokalen Modellausschnitt nicht mehr benötigt werden, in den formalen Objekten des Operationsoutputs auch nicht mehr berücksichtigt werden. Auf den ersten Blick entspricht dies einer "effizienten" Modellierungsweise, die alle nicht mehr relevanten Informationen über einen Realitätsausschnitt unterdrückt. Diese lokale Informationsvernichtung führt aber zu unerwünschten *Fernwirkungen*, falls in anderen Modellausschnitten die zuvor vernachlässigten Informationen über das modellierte reale Objekt doch wieder benötigt werden. Denn dann stehen die benötigten, aber zuvor vernichteten Informationen nicht mehr zur Verfügung. Es erfolgte eine vorschnelle Modelloptimierung unter Effizienzaspekten zu Lasten der Modellierungsgüte[37].

Die Gefahr struktureller Integritätsverletzungen in Prädikat/Transition-Netzen wird anhand eines Beispiels verdeutlicht. Es betrifft eine ungewollte Fernwirkung in der Gestalt eines modellierungsbedingten, aber unerwünschten Informationsverlusts. Die Integritätsverletzung beruht auf der Beliebigkeit der Zusammenfassung von "Individuen" zu Komplexeinheiten - einschließlich der reziproken Auflösung solcher Komplexeinheiten in ihre Bestandteile. Beispielsweise läßt sich ein Werkstück als ein K-Tupel von "Individuen" modellieren, die jeweils eine Eigenschaft des Werkstücks repräsentieren. Bei der Verwendung von Prädikat/Transition-Netzen ist es üblich, die Ein- und Ausgangskanten einer Transition nur mit K-Tupeln aus solchen "Individuen" zu beschriften, von der das Schaltverhalten der Transition abhängt[38]. Falls eine Transition die Bearbeitung des Werkstücks modelliert, kann es sich bei solchen relevanten Eigenschaften beispielsweise um Materialeigenschaften des Werkstücks oder um seine Bearbeitungspriorität handeln. Andere Werkstückeigenschaften ohne Bearbeitungsrelevanz werden dagegen als Anschriften der Transition nicht berücksichtigt. Hierbei könnte es sich z.B. um das Werkstückgewicht handeln, das erst bei einem Werkstücktransport Bedeutung erlangte. Aufgrund der prinzipiell lokalen Definition konventioneller Petrinetze - zu denen hier auch die Prädikat/Transition-Netze gerechnet werden - existiert jedoch kein globaler Speicher, in dem die Informationen über solche Werkstückeigenschaften vorgehalten werden könnte.

Das Außerachtlassen von Objekteigenschaften an einer Transition, an der diese Eigenschaften lokal irrelevant sind, hätte daher die fatale Folge, daß das Wissen über solche irrelevanten Eigenschaften verloren ginge[39)]. Dann kann aber der Fall eintreten, daß das betroffene Werkstück die betrachtete Transition passiert und an einer nachfolgenden Transition, die etwa den Werkstücktransport abbildet, die vormals unwesentliche Werkstückeigenschaft schaltrelevant wird. Aufgrund des Informationsverlusts kann auf diese Werkstückeigenschaft im Netzmodell aber nicht mehr zugegriffen werden. Um solche Fernwirkungen zu vermeiden, müßte sich der Modellgestalter stets vergegenwärtigen, welche Eigenschaften eines modellierten Objekts an einer Transition zwar lokal ohne Belang bleiben, aber im weiteren Objektfluß wichtig werden könnten. Es wäre erforderlich, solche Objekteigenschaften als invariante "Individuen" in den Kantenanschriften der jeweils modellierten Transition zu berücksichtigen.

Der Verf. hält diese Anforderung aus der Sicht der praktischen Modellgestaltung für inakzeptabel. Bei der Modellierung komplexer realer Systeme dürfte die Aufmerksamkeitsspanne der Modellgestalter oftmals überfordert sein, um alle potentiellen Fernwirkungen dieser Art zu überblicken. Darüber hinaus würde der grundsätzlich begrüßenswerte Aspekt des Petrinetz-Konzepts, die Dynamik komplexer Systeme durch Schaltakte rein lokal definierter Transitionen zu modellieren und hierdurch die Komplexität der Modellgestaltung erheblich zu reduzieren, in sein Gegenteil verkehrt. Daher wird der Verf. eine Definition von Marken einführen, die stets die Attributstruktur der jeweils modellierten realen Objekte unabhängig von der lokalen Schaltcharakteristik einer Transition erhalten. Auf diese Weise bleiben Informationen über Objekteigenschaften erhalten, ohne daß sich der Modellgestalter bei seinen lokalen Modellierungsentscheidungen um problematische Fernwirkungen der zuvor skizzierten Art kümmern müßte. Zugleich wird der Modellgestalter durch die spezielle Implementierungsweise Synthetischer Netze auch von der mühsamen Spezifikation aller lokal irrelevanten Attributinformationen entlastet[40)].

Um die Modellierungsschwierigkeiten der voranstehend skizzierten Art erst gar nicht entstehen zu lassen, wurde oben die strukturelle Integritätsbedingung aufgestellt. Durch ihr Strukturerhaltungspostulat begrenzt sie die eingangs angeführte formale Freizügigkeit bei der Strukturierung formaler Objekte. Die entsprechenden Einschränkungen werden speziell für Modellierungen, die auf der Basis von Synthetischen Netzen erfolgen, in einer markenbezogenen, prädikatenlogisch erweiterten Signatur formal verankert. Dieser spezielle Signaturansatz wird auch kurz als Markensignatur MSIG bezeichnet. Er wird aus dem früher allgemein definierten, prädikatenlogisch erweiterten Signaturkonzept abgeleitet.

Ausgangspunkt für die Entfaltung einer Markensignatur MSIG ist eine prädikatenlogische Signatur PSIG=(SO,OP,PRÄ). Sie wird in formaler Hinsicht unverändert übernommen. Eine Markensignatur ist daher ebenso als ein Tripel MSIG=(SO,OP,PRÄ) definiert. Allerdings wird die formale Kombinationsvielfalt der Sorten, Operations- und Prädikatssymbole aus den Familien SO, OP bzw. PRÄ in materieller Hinsicht eingeschränkt. Denn nicht jede formal zulässige prädikatenlogische Signatur PSIG stellt eine materiell zulässige Markensignatur MSIG dar. Als materielles Selektionskriterium dient eine zentrale ontologische Modellierungsprämisse. Sie konstituiert eine Markenontologie für Netzmodelle[41)]. Diese Prämisse unterstellt, daß jedes reale Objekt aus dem jeweils modellierten Realitätsausschnitt im zugehörigen Netzmodell durch die Kopie einer Marke[42)] repräsentiert wird[43)]. Diese Marken stellen eine erste fundamentale Kategorie formaler Objekte von Netzmodellen dar[44)].

Eine weitere Basisprämisse dieser Ausarbeitung besteht in der Annahme, daß sich *alle* modellierungsrelevanten Informationen über reale Objekte mit der Hilfe von zwei Informationsmodi auszudrücken lassen: Es handelt sich entweder um Informationen über die Ausprägungen von Eigenschaften (Attributen)[45] eines realen Objekts[46]. Oder die Informationen erstrecken sich auf die Existenz eines realen[47] Objekts[48]. Der erste Informationsmodus wird in Netzmodellen durch strukturierte Marken abgedeckt, die sich vollständig auf Attributmarken zurückführen lassen. Dem zweiten Informationsmodus wird die strukturlose Basismarke gerecht. Beide Markenarten werden auf der Basis des Signaturkonzepts formal definiert und unter dem Begriff der sortierten Marken zusammengefaßt.

Die zweite fundamentale Kategorie formaler Objekte, die in Netzmodellen definiert sind, stellen die Attributausprägungen dar. Dabei handelt es sich um die Ausprägungen derjenigen Attribute, die zuvor als Eigenschaften von modellierten realen Objekten eingeführt worden sind. Jede Attributausprägung repräsentiert als formales Objekt eine Information über die entsprechende reale Eigenschaftsausprägung. Andere Kategorien fundamentaler formaler Objekte als die beiden vorgenannten Kategorien der Marken und Attributausprägungen sind für Netzmodelle nicht definiert.

Auf der Grundlage der voranstehenden Festlegungen läßt sich die strukturelle Integritätsbedingung für die formalen Objekte eines Netzmodells mit Hilfe einer markenbezogenen algebraisch-prädikatenlogischen Spezifikation auf der Grundlage des Signaturkonzepts präzisieren. Dabei wird das Strukturerhaltungspostulat der Integritätsbedingung allerdings nicht - wie andere Integritätsbedingungen - als prädikatenlogische Formel ausgedrückt. Vielmehr äußert sich seine fundamentale ontologische Bedeutung darin, daß es die gesamte Konstitution derjenigen formalen Objekte prägt, die für ein Synthetisches Netz und jedes darauf aufbauende Netzmodell zulässig sind. Diese Objektkonstitution erfüllt die strukturelle Integritätsbedingung *per constructionem.* Für sie gelten folgende ontologischen Postulate:

- ❑ Jedes reale Objekt wird ausschließlich durch formale Objekte der ersten Kategorie - die Kopien von Marken - repräsentiert.
- ❑ Jede Eigenschaftsausprägung eines realen Objekts wird durch genau ein formales Objekt der zweiten Kategorie - eine Attributausprägung - dargestellt.
- ❑ Andere formale Objektkategorien als die der Markenkopien und der Attributausprägungen gibt es nicht.
- ❑ Die formalen Objekte, die Markenkopien[49] oder Attributausprägungen darstellen, können sowohl atomar als auch beliebig komplex zusammengesetzt sein.
- ❑ Marken und Attribute, die den Markenkopien bzw. Attributausprägungen zugrundeliegen, werden jeweils als Sorten definiert. Die Gruppen der marken- und der attributspezifischen Sorten sind disjunkt.
- ❑ Die Basismarke ist die einzige Marke, deren Kopien jeweils atomare formale Objekte darstellen. Bei den Kopien aller anderen Marken handelt es sich um zusammengesetzte formale Objekte.
- ❑ Jede Kopie einer Attributmarke wird durch mindestens[50] eine markenspezifische Erzeugungsoperation aus Attributausprägungen zusammengesetzt.
- ❑ Jede Kopie einer Kompositmarke wird durch mindestens eine markenspezifische Kompositionsoperation aus Attributmarkenkopien oder[51] aus bereits gebildeten Kompositmarkenkopien in beliebig komplexer Weise zusammengesetzt.
- ❑ Jede Attribut- oder Kompositmarke heißt eine strukturierte Marke. Ihre Struktur wird durch Attribute bzw. andere Marken konstituiert. Analoges gilt für die jeweils zugehörigen Markenkopien.

- Die Basismarke kann an der Konstitution keiner strukturierten Marke teilnehmen. Daher können Kopien der Basismarke auch nicht in den Kopien von strukturierten Marken enthalten sein.
- Die Basismarke und alle strukturierten Marken bilden die Gesamtheit aller sortierten Marken. Andere Marken gibt es nicht.
- Die Argumente von Prädikatssymbolen können sich unmittelbar nur auf sortierte Marken beziehen. Die Argumente von prädikatenlogischen Formeln dürfen unmittelbar nur aus Kopien von sortierten Marken bestehen.
- Infolge der Zusammensetzung von Attributmarken aus Attributen können sich Prädikatssymbole und prädikatenlogische Formeln mittelbar auch auf Attribute bzw. Attributausprägungen erstrecken.
- Die teilevaluierten Terme aus Restriktionsformeln dürfen sowohl Markenkopien als auch Attributausprägungen[52)] darstellen. Ein Drittes ist unzulässig.

Durch diese Festlegungen einer Markenontologie unterscheidet sich die markenbezogene Konzipierung Synthetischer Netze deutlich von den Prädikat/Transition-Netzen. Die gleiche Abgrenzung gilt in bezug auf die wenigen bisher vorgelegten Ansätze, Petrinetze auf der Basis des algebraischen Signaturkonzepts zu entwickeln. Die dort verwendeten SIG-Spezifikationen kennen die oben vorgestellten ontologischen Postulate für die Konstitution formaler Objekte in Netzmodellen nicht. Diese Postulate werden aber im Rahmen von Markensignaturen berücksichtigt. Dies wird anschließend näher ausgeführt.

Anmerkungen zum Kapitel:

1) Vgl. zu Prädikat/Transition-Netzen NIEHUIS (1986), S. 17ff.; GENRICH (1987a), S. 212ff.; THIELER-MEVISSEN (1987b), S. 33ff.; GENRICH (1988b), S. 229ff., insbesondere S. 234f.; MURATA,TA. (1988b), S. 483f.; PAGNONI (1990), S. 25ff. u. 156ff.; ABEL,D. (1990), S. 39 u. 41f.; ROSENSTENGEL (1991), S. 58ff.

2) Vgl. zu Gefärbten Netzen JENSEN (1987a), S. 249ff.; VAUTHERIN (1987b), S. 299ff.; FREEDMAN (1988b), S. 331; ABEL,D. (1990), S. 38ff. u. 42.

3) Gefärbte Netze und Prädikat/Transition-Netze verhalten sich strenggenommen nur so lange äquivalent, wie sich die Menge aller beweglichen Objekte in einem Höheren Netz durch eine *endliche* Anzahl verschiedenartiger Marken darstellen läßt. Diese Voraussetzung ist erforderlich, weil in einem Gefärbten Netz immer nur eine *endliche* Anzahl unterschiedlicher Markenfarben definiert ist; vgl. ABEL,D. (1990), S. 38 u. 42. Wenn die Finitheitsprämisse erfüllt ist, spielt es keine Rolle, ob die endlich vielen Marken entweder durch die "gefärbten" Marken eines Gefärbten Netzes oder aber durch die "individuellen" Marken eines Prädikat/Transition-Netzes erfaßt werden. Andernfalls - wenn unendliche viele verschiedenartige Marken zugelassen sein sollen - können Gefärbte Netze nicht mehr angewendet werden. Die potentielle Unendlichkeit der Artenvielfalt von Marken bereitet dagegen bei Prädikat/Transition-Netzen keine Schwierigkeiten. Denn sie sind so "flexibel" ausgelegt, daß sie die Vorgabe einer endlichen Menge von verschiedenartigen Marken nicht erfordern; vgl. ABEL,D. (1990), S. 39. Der tiefere Grund für diese Divergenz im infiniten Fall liegt in der unterschiedlichen Weise, in der "gefärbte" und "individuelle" Marken in den arithmetischen Kalkülen der Gefärbten Netze bzw. der Prädikat/Transition-Netze gehandhabt werden. ABEL,D. (1990), S. 38f., hat dies anhand von zwei instruktiven Inzidenzmatrizen veranschaulicht. Sie erfassen die gleiche Netztopologie einmal als ein Gefärbtes Netz und das andere Mal als ein Prädikat/Transition-Netz (auf S. 40f. folgt ein analoges Beispiel für eine verdichtete Netztopologie).
Die voranstehende Erläuterung bedarf einer dreifachen Kommentierung. Erstens erstreckt sich die Finitheitsvoraussetzung Gefärbter Netze nur auf die Definition verschiedenartiger Marken. Sie schließt dagegen keineswegs aus, daß von der gleichen Marke im selben Netz beliebig viele Kopien vorhanden sind. Im Grenzfall kann es sich auch um unendlich viele Markenkopien handeln, sofern Stellen mit unbeschränkten Markenkapazitäten vorkommen. Zweitens sieht der Verf. in der "flexiblen" Auslegung der Prädikat/Transition-Netze, die auch unendlich viele verschiedenartige Marken zuläßt, keinen besonderen Vorzug. Vielmehr reflektiert sie nur die mangelhafte Verankerung des formalen Konstrukts "Marke" in der Definition von Prädikat/Transition-Netzen. Darauf wird in diesem Kapitel ausführlich eingegangen.
Drittens wird die spezielle Finitheitsvoraussetzung der Gefärbten Netze von den Synthetischen Netzen nicht geteilt. Dies mag auf den ersten Blick erstaunen, weil auch in dieser Arbeit des öfteren ähnlich anmutende Endlichkeitsforderungen aufgestellt werden. Diese Endlichkeitsforderungen stellen aber im wesentlichen nur sicher, daß die Erreichbarkeitsgraphen von Netzen endlich groß ausfallen. Dagegen dürfen die Attributmarken von Synthetischen Netzen durchaus unendliche Definitionsbereiche für ihre zulässigen Attributausprägungen besitzen. Jede Kombination der Ausprägungen aller Attribute einer Attributmarke, die sich in einer Kopie dieser Attributmarke manifestiert, müßte in einem Gefärbten Netz durch eine kombinationsspezifisch "gefärbte" Marke erfaßt werden. Schon ein unendlicher Definitionsbereich für mindestens eines der involvierten Attribute führt dazu, daß unendlich viele verschiedene Attributausprägungskombinationen gebildet werden können. Folglich wären in einem Gefärbten Netz unendlich viele kombinationsspezifisch "gefärbte" Marken erforderlich, um alle Kopien der betrachteten Attributmarke wiedergeben zu können (q.e.d.). Daher kann die o.a. Äquivalenzbehauptung für Gefärbte Netze und Prädikat/Transition-Netze in bezug auf Synthetische Netze strenggenommen nicht aufrechterhalten werden. In der anschließenden Argumentation zur Abgrenzung zwischen Synthetischen Netzen und "typischen" Höheren Netzen spielt die potentiell unendliche Kombinationsvielfalt der Attributausprägungen von Attributmarken jedoch keine Rolle. Daher kann von ihrem Einfluß auf die Differenzierung zwischen Gefärbten Netzen und Prädikat/Transition-Netzen abgesehen werden. Vor dem Hintergrund dieser Vereinfachung wird oben davon gesprochen, daß sich die beiden letztgenannten Netzklassen "kaum noch" unterscheiden.

4) Die Auswahl von Prädikat/Transition-Netzen erfolgt keineswegs willkürlich. Denn sie knüpfen an das prädikatenlogische Fundament, das den Modellierungen dieser Arbeit zugrundeliegt, unmittelbar an. Gefärbte Netze besitzen zwar im Prinzip ein gleichwertiges prädikatenlogisches Ausdrucksvermögen, solange von ihrer speziellen Finitheitsvoraussetzung abgesehen werden kann (vgl. die voranstehende Anmerkung). Aber die prädikatenlogische Formulierungskraft wird bei Gefärbten Netzen weniger deutlich. Daher wird in den hier vorgelegten Untersuchungen auf Gefärbte Netze nur dann ausdrücklich eingegangen, wenn der Aspekt ihrer Marken- oder Schaltfarben im Vordergrund steht.
Allerdings räumt der Verf. ein, daß ihn ausschließlich sein Interesse an der prädikatenlogischen Formulierung von Problemrepräsentationen dazu bewogen hat, Prädikat/Transition-Netze zu bevorzugen. Dies entspricht dem primären Interesse dieser Arbeit, ausdrucksmächtige Konzepte für die Konstruktion von problemrepräsentierenden Modellen zu erforschen. Aus der Perspektive der - hier hintenan gestellten - Lösungsorientierung stimmt er jedoch ohne Einschränkungen der Ansicht ABEL's zu, daß sich Gefärbte Netze infolge ihrer speziellen Finitheitsvoraussetzung wesentlich einfacher auswerten lassen (vgl. ABEL,D. (1990), S. 41f.). Er verdeutlicht anhand eines ein-

fachen, aber überzeugenden Beispiels (S. 40f.), daß gerade die "flexible" Verwendung von Variablen und Funktionen in den Koeffizienten der Inzidenzmatrizen von Prädikat/Transition-Netzen erhebliche rechentechnische Schwierigkeiten bereitet. Gefärbte Netze kommen dagegen bei gleichen Netztopologien mit Inzidenzmatrizen aus, deren Koeffizienten lediglich aus einfach handhabbaren konstanten Ganzzahlen bestehen. (Dies gilt allerdings nur, solange die Finitheitsvoraussetzung erfüllt ist). Vgl. dazu auch das Beispiel, das in einer der voranstehenden Anmerkungen illustrierte, wie sich Höhere Netze besonders kompakt darstellen lassen.

5) Darüber hinaus weichen Synthetische Netze in weiteren Aspekten von Prädikat/Transition-Netzen ab, die jedoch im Rahmen dieser Arbeit keine Rolle spielen. Dabei handelt es sich vor allem um die "statischen" - d.h. hinsichtlich ihrer Extension unveränderlichen - Formeln, die einem Prädikat/Transition-Netz insgesamt als dessen "Unterstützung" (support) zugeordnet sind. Ebenso betroffen sind die Selektorformeln für die Stellen, Transitionen und Kantengewichte von Prädikat/Transition-Netzen. Vgl. zu den vorgenannten Konstrukten GENRICH (1988b), S. 234f., und das Beispielnetz in Abb. 2 auf S. 236. Die Selektorformeln der Transitionen und Kantengewichte eines Prädikat/Transition-Netzes lassen sich in einem Synthetischen Netz ohne Schwierigkeiten durch Makrotransitionen verwirklichen, deren Schaltvorschriften Produktionsregeln umfassen. Dabei entspricht jeder Selektionsbedingung aus einer Selektorformel die Antezedensbedingung von genau einer Produktionsregel. Die Selektorformeln der Stellen eines Prädikat/Transition-Netzes können dadurch berücksichtigt werden, daß zunächst die Transformation benutzt wird, die bei GENRICH (1988b), S. 244 (4. Absatz) u. 245 (Abb. 7), beschrieben wird. Sie erzeugt ein äquivalentes Prädikat/Transition-Netz, dessen Stellen den immer gültigen, tautologischen Selektor besitzen. Sie lassen sich als selektorfreie Stellen behandeln. Schließlich ist es möglich, die statischen Formeln der Netzunterstützung in Synthetischen Netzen als Restriktionsformeln wiederzugeben. Sie können entweder jeder Transition zugeordnet werden oder aber auf diejenigen Transitionen beschränkt werden, deren Schaltverhalten sie restriktiv zu beeinflussen vermögen. Infolge der voranstehend skizzierten Transformationsmöglichkeiten fällt das Ausdrucksvermögen Synthetischer Netze nicht wesentlich geringer als das von Prädikat/Transition-Netzen aus. Lediglich wird auf die Option besonders kompakter Netzdefinitionen verzichtet, die sich bei der Verwendung von Stellen-, Transitionen- und Kantenselektoren sowie Netzunterstützungen eröffnen. Solche Netzkompaktifizierungen erlangen aber in den hier vorgelegten Untersuchungen keine Bedeutung.

6) Vgl. zur grundsätzlichen Bedeutung *ontologischer Prämissen (Prinzipien)* für die Ausgestaltung formaler Konzepte LEE,R. (1988a), S. 221 u. 225f.; WAND (1989), S. 538ff., insbesondere S. 539 (dabei lehnt sich WAND explizit an die ontologischen Prinzipien an, die sich bei BUNGE (1977), S. 16f., 28ff., 72ff., 111, 119f., 126ff., 140ff., 218ff., 255 u. 259f., finden); VON LUCK (1989), S. 281ff. Diese Prämissen legen fest, welche Kategorien formaler Objekte in einem Modellierungskonzept für die Repräsentation von Objekten aus der realen Welt grundsätzlich vorgehalten werden. Dabei werden jedoch - im Gegensatz zu denotationalen Semantiken - einzelnen formalen Objekten noch keine konkreten Referenzen auf bestimmte reale Objekte zugeordnet. So spricht auch VON LUCK (1989), S. 281, von "allgemeinen Kategorien ..., die gegenstandsunabhängig den prinzipiellen Weltaufbau angeben sollen." Die formalen Objektkategorien werden in dieser Arbeit auf der Grundlage des Signaturkonzepts durch die jeweils definierten Sorten festgelegt. Im Entity-Relationship-Konzept, das im voranstehenden Kapitel erwähnte wurde, erfolgt eine entsprechende Kategoriendefinition durch die Entity-Typen.

7) Dieser Unterschied führt dazu, daß Synthetische Netze - unter dem Aspekt der Markenstrukturierung - eine Teilklasse der Prädikat/Transition-Netze darstellen. Denn es werden nicht alle in Prädikat/Transition-Netzen zulässigen Marken in Synthetischen Netzen erlaubt, sondern nur solche Marken, die sich als Repräsentanten von Objekten interpretieren lassen. Zugleich stellen Synthetische Netze aber unter anderen Gesichtspunkten - wie etwa hinsichtlich der Zulässigkeit von Nulltests oder nicht-konjunktiver Kantenlogiken - auch Erweiterungen von Prädikat/Transition-Netzen dar, für die derartige Netzkomponenten nicht definiert sind.

8) Der Übersichtlichkeit halber bezieht sich die nachfolgende Argumentation explizit nur noch auf Attributmarken. Implizit gilt sie jedoch ebenso für die daraus abgeleiteten Kompositmarken.

9) Ein Objekt wird strenggenommen stets durch die Kopie einer Attribut- oder Kompositmarke repräsentiert. Der Einfachheit halber wird hier aber nur von Marken gesprochen. Dies wurde bereits vereinbart.
Infolge des Realproblembezugs gehen die nachfolgenden Anmerkungen über eine syntaktische Betrachtung hinaus. Sie setzen eine Netzinterpretation durch eine denotationale Semantik voraus, die erst später präzisiert wird. Trotz seiner früheren Ausgrenzung aus der syntaktischen Erweiterungsperspektive von Stelle/Transition-Netzen wird der denotationale Aspekt hier kurz gestreift. Denn nur er rechtfertigt die besondere Qualität der *formalen* Markendefinition in Synthetischen Netzen, die von der üblichen Definitionsweise der Prädikat/Transition-Netze abweicht.

10) Dies gilt allerdings nur für die hier angesprochenen Attributmarken als atomare strukturierte Marken. Die zusammengesetzten strukturierten Marken stellen dagegen per constructionem keine minimalen Einheiten dar, sondern können in ihre atomaren Komponenten zerlegt werden.

11) Ein ähnliches Objektverständnis liegt der Simulationssprache SIMAN und ihrer Animationskomponente CINEMA zugrunde. Dort werden "bewegliche Systemelemente" oder "Entities" als unteilbare dynamische Objekte behandelt, die sich durch ein simuliertes System bewegen können. Dabei ist es möglich, sowohl Objekteigenschaf-

ten zu verändern als auch durch die Objekte Veränderungen in ihren jeweils simulierten Umsystemen zu veranlassen. Vgl. zu dieser ontologischen, am Objektbegriff anknüpfenden Basis von SIMAN und CINEMA TEMPELMEIER,H. (1989a), S. 30; SYSTEMS MODELING CORPORATION (o.J.b), S. 6.
Allerdings bleibt der Objektbegriff des voranstehenden Simulationskonzepts auf bewegliche Objekte beschränkt. Die Objektbewegung in einem simulierten und graphisch animierten System entspricht zwar dem Fluß von Marken durch ein Petrinetz. Doch ist die Markenontologie des hier entfalteten Petrinetz-Konzepts breiter angelegt. Sie bleibt nicht auf den Aspekt *beweglicher* Objekte eingeschränkt. Vielmehr können Marken ebenso reale Objekte repräsentieren, die keine raumzeitlichen Bewegungen auszuführen vermögen. Dabei kann es sich z.B. um strukturierte Marken handeln, die Informationen darstellen und immer auf derselben Stelle eines Netzes verharren. Dennoch lassen sich durch solche ortsfesten Marken Veränderungen ausdrücken: Die Variation der Ausprägungen von Markenattributen bildet zeitlich variable Informationsinhalte ab. Ein ähnlich weit gefaßter Objektbegriff liegt den "Entities" zugrunde, die KOCHAN,D. (1986), S. 133, als Basiskonzepte für diskrete Simulationsstudien verwendet.

12) Strenggenommen handelt es sich um die Attributausprägungen, die an der Konstitution einer Markenkopie teilhaben. Es entspricht aber der bereits eingeführten Diktionsvereinfachung, Markenkopien kurz als Marken anzusprechen, die Attributausprägungen analog als Attribute zu thematisieren.

13) Damit wird vom objektorientierten Ansatz dezidiert abgewichen. Dort werden die Ausprägungen von Objektattributen ebenso als Objekte behandelt. Besonders deutlich wird dies bei BANCILHON (1986), S. 53. Daher findet aus objektorientierter Sicht die ontologische Differenz zwischen real existierenden Objekten einerseits und objektbeschreibenden Attributausprägungen andererseits überhaupt keine Berücksichtigung.

14) Vgl. z.B. MURATA,TA. (1988b), S. 484.

15) Mitunter werden diese Individuen nicht explizit als Individuen angesprochen, sondern unter inhaltlich äquivalenten Bezeichnungen thematisiert. Beispielsweise spricht MURATA,TA. (1988b), S. 483f., von Konstanten und Variablen. Diese stellen aber "Individuen" im Sinne des prädikatenlogischen Kalküls dar, der später im Detail entfaltet wird.

16) Vgl. zum folgenden die Beschreibungen von Prädikat/Transition-Netzen in denjenigen Quellen, die bereits zu dieser Netzklasse aufgeführt wurden.

17) Es ist deutlich darauf hinzuweisen, daß die nachfolgenden kritischen Anmerkungen zum Individuenbegriff der Prädikat/Transition-Netze sich nur auf den pragmatischen Kontext beziehen, in dem Netze zur Modellierung von Realproblemen herangezogen werden. Nur in diesem Zusammenhang können Schwierigkeiten entstehen, falls die "Individuen" aus Prädikat/Transition-Netzen im Sinne von individuellen Objekten aus Beschreibungen von Realproblemen interpretiert werden. Außerhalb dieses pragmatischen Kontextes erweist sich der Individuenbegriff von Prädikat/Transition-Netzen jedoch als unproblematisch. "Individuen" sind - aus syntaktischer Perspektive - identisch mit Elementen aus den Definitionsbereichen der Variablen von Prädikaten. Sie stellen somit logische Konstanten, d.h. konstante Zeichen aus einem formalsprachlichen Alphabet, dar. Um den realproblem- und modellbezogenen Begriff individueller Objekte vom formallogischen Individuenbegriff zu differenzieren, werden Individuen im letztgenannten Sinne durch Anführungszeichen gekennzeichnet.

18) Das "Individuum" des Grenzfalls K=0 ist das Nulltupel "()" = ∅ für die Basismarke.

19) Vgl. GENRICH (1988b), S. 229 ("tokens"); OBERWEIS (1988a), S. 5f.

20) Vgl. GENRICH (1987a), S. 216f.; MURATA,TA. (1988b), S. 483f.
Einen Mittelweg schlägt PAGNONI (1990), S. 27, 158 u. 160, ein. Sie verwendet den Tupelbegriff, bezieht sich aber ebenso auf den Markierungsbegriff.
Eine weitere Bezeichnungsvariante findet sich bei MURATA,TA. (1988b), S. 484, und PAGNONI (1990), S. 27 u. 158: Dort werden die K-Tupel auch als "Farben" (color) oder als "Farbmengen" (color sets) angesprochen. Diese Benennung wird hier nicht weiter verfolgt, da sie mit den - anders definierten - Schaltfarben von Transitionen verwechselt werden könnten.

21) Vgl. z.B. GENRICH (1987a), S. 228, Figure 11, als Demonstration solcher beliebigen Umgruppierungen von "Individuen" (Attributen) in den n-Tupeln (Marken) von Prädikat/Transition-Netzen.

22) Diese ontologische Indifferenz wird hinsichtlich relationaler Datenschemata von MAYR,H. (1987), S. 503f., klar herausgearbeitet. Dabei wird zwischen "starken" und "weichen Gegenständen" differenziert, die jeweils eigenständig existieren können bzw. nur im Zusammenhang mit anderen Gegenständen betrachtet werden.

23) Allenfalls könnten mehrere Kopien desselben "Individuums" multiple Sichtweisen auf dieses Individuum repräsentieren. Diese Möglichkeit wurde bereits in einer früheren Anmerkung angesprochen. Dort wurde allerdings auch darauf hingewiesen, daß diese Option in Beiträgen zur Prädikat/Transition-Netzen keine explizite Würdigung erfährt.

24) Vgl. z.B. OBERWEIS (1988a), S. 6, mit der Formulierung: "Solche Individuen können auch ... komplexe Struktur besitzen, d.h. ihre Komponenten sind selbst wieder Individuen ...". Die erstgenannten "Individuen" beziehen sich auf Marken, die zweitgenannten dagegen auf Markenattribute.

25) Zwar werden bei der Definition von Prädikat/Transition-Netzen durchaus zusammengesetzte Terme zugelassen; vgl. z.B. GENRICH (1988b), S. 231. Mit ihrer Hilfe könnten hierarchische Attributstrukturen definiert werden. Aber diese formale Ausdrucksmöglichkeit wird nicht genutzt. Statt dessen werden mitunter alle zusammengesetzten Terme aus der Definition von Prädikaten explizit ausgeschlossen; vgl. abermals GENRICH (1988b), S. 234f.

26) Solche Attributhierarchien finden sich beispielsweise bei BANCILHON (1986), S. 54; OBERWEIS (1988a), S. 6f.

27) Vgl. zu solchen rekursiven Objektkompositionen wiederum BANCILHON (1986), S. 53f. Sie lassen sich ebenso als iterative Zusammensetzung von Objekten auffassen. Bei der rekursiven Bezeichnungsweise wird gedanklich von den komplexeren zu den einfacheren Objekten zurückgeschritten. Der iterativen Alternative liegt dagegen die Perspektive zugrunde, aus einfacheren Objekten komplexere aufzubauen.

28) Durch diese explizite Einführung aller Marken eines Netzes als Basiskonstrukte wird eine besondere Schwierigkeit der programmiertechnischen Behandlung dynamischer Objekte vermieden. Denn das "dynamische Erzeugen" von Objekten, die nicht ex ante definiert wurden, läßt sich in Programmiersprachen für die Netzimplementierung in der Regel nicht - oder zumindest nur unter beträchtlichen Schwierigkeiten - realisieren. Vgl. hierzu beispielsweise die Anmerkungen bei STEINMETZ,RA. (1987a), S. 526f., zur dynamischen Objekterzeugung in der Programmiersprache OCCAM. (Diese Programmiersprache eignet sich infolge ihrer inhärenten Parallelität grundsätzlich für die Implementierung von Netzen mit nebenläufigem Schaltverhalten. Sie wird daher - wie auch in der vorgenannten Quelle - zunehmend für die Netzimplementierung angewendet.) Vgl. ebenso die Ausführungen von WENDT (1989), S. 230ff., zur Modellierung von Systemen, in denen einzelne Strukturkomponenten im Zeitablauf entstehen oder auch wieder verschwinden können. Obwohl sich der Verf. den Überlegungen von WENDT inhaltlich keineswegs immer anschließen möchte, vermitteln sie jedoch einen Einblick in die Schwierigkeiten, dynamische Objekterzeugungen oder -vernichtungen zu modellieren. Durch den Ansatz Synthetischer Netze, alle modellierungsrelevanten Marken von vornherein als Basiskonstrukte explizit einzuführen, entfällt hingegen die Notwendigkeit, Objekte unter Umständen dynamisch generieren zu müssen. Dynamische Objektvernichtungen werden ebensowenig erforderlich. Denn die Markendefinitionen gelten auch dann noch, wenn in einem Netzmodell alle Kopien einer Marke untergegangen sind.
Die initiale Deklaration aller zulässigen Marken eines Synthetischen Netzes bedeutet allerdings nicht, daß reale Prozesse der Objekterzeugung oder -vernichtung nicht modelliert werden können. Denn die initiale Markendeklaration legt nur das *Potential* der grundsätzlich definierten *Marken* fest. Welche Marken*kopien* in einem aktuellen Zustand der Netzmodells tatsächlich existieren und ob einzelne *Kopien* dieser Marken bei schaltbedingten Übergängen zwischen benachbarten Netzmarkierungen erzeugt oder vernichtet werden, wird durch die Markendeklaration der Netzdefinition dagegen in keiner Weise beeinflußt. Auch DORN (1989), S. 67f., weist darauf hin, daß bei der Repräsentation von Wissen über Prozeßkoordinierungen die Erzeugbarkeit und Vergänglichkeit von Objekten berücksichtigt werden müssen: Es komme auf die Fähigkeit an, die (aktuelle) Existenz von Objekten darzustellen zu können.
Im Rahmen der hier vorgelegten Markenontologie bereitet es keine Schwierigkeiten, Beginn und Ende einer Objektexistenz durch entsprechende Erzeugungs- bzw. Vernichtungsakte zu erfassen. Zu diesem Zweck können einfache, übersichtliche Netzkonstrukte verwendet werden. Vgl. dazu den Objektgenerator und den Objektabsorber aus der nachfolgenden Abb. 50. Beide Konstrukte beziehen sich auf eine bereits deklarierte, beliebige sortierte Marke der Art "s". Die Kopien "m_s" dieser Markenart repräsentieren jeweils ein Objekt. Durch einen Schaltakt der Generatortransition t_G wird auf ihrer Ausgangsstelle s_1 eine Markenkopie "m_s" abgelegt. Dies entspricht einer Objekterzeugung. Das Schalten der Absorbertransition t_A zieht dagegen eine Markenkopie "m_s" von ihrer Eingangsstelle s_1 ab. Das spiegelt eine Objektvernichtung wider. In beiden Fällen dient die eine Kopie "∅" der Basismarke, die sich auf der Stelle s_2 befindet, nur dem technischen Zweck anzuzeigen, daß die Generator- oder Absorbertransition ihre generative bzw. destruktive Funktion erfüllen kann. Die Notation "<...>" der Kantenanschriften spielt hier keine Rolle. Sie wird später bei der Vorstellung Synthetischer Netze näher erläutert. Darüber hinaus ist zu beachten, daß die hier vorgestellten Absorbertransitionen konventionell definierte Netzkanten besitzen. Sie stehen daher in keiner Beziehung zu den besonderen Absorberkanten.
Die beiden voranstehend skizzierten Netzkonstrukte für das explizite Erzeugen oder Vernichten von Objekten reichen über die übliche Ausdruckskraft von objektorientierten Gestaltungsansätzen hinaus. Denn jene Ansätze kennen keine analogen Instrumente; vgl. WAND (1989), S. 549 ("There are no explicit mechanisms for the creation or destruction of things.") u. 557.

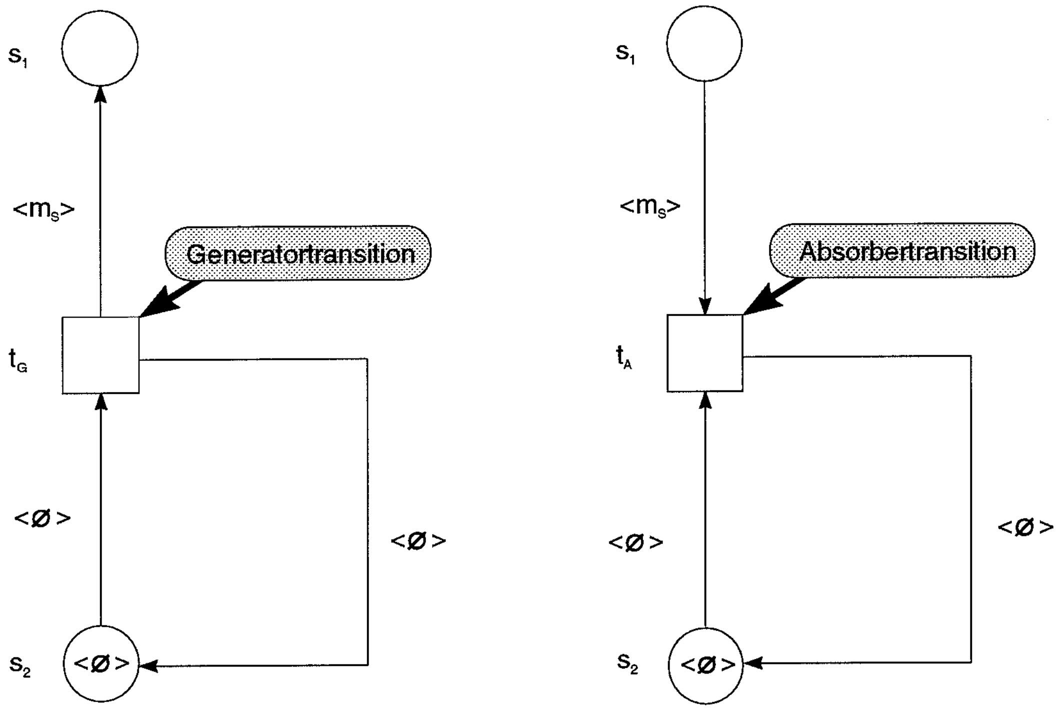

Abb. 50: Generator- und Absorbertransitionen

29) In dieser Hinsicht ist ESCHENBACHER (1991), S. 223, beizupflichten, wenn er kritisiert: "Die Marken, die in Petri-Netzen verschoben werden, suggerieren, daß diese bewegliche Teile repräsentieren könnten. Das ist aber nicht der Fall. ... Im Gegensatz zu physikalischen Teilen können sie sich teilen oder verschmelzen." In *anderer* Hinsicht kann die Einlassung ESCHENBACHER's aber nicht akzeptiert werden. Darauf wird an anderer Stelle näher eingegangen.

30) Der Persistenzbegriff wird hier in Anlehnung an das "Law of Persistence" gebildet, das von GEORGEFF (1986), S. 72f., thematisiert wird. Vgl. ebenso das Hilfsprädikat "persist" bei DORN (1989), S. 28, das die zeitlich andauernde Gültigkeit eines Sachverhalts ausdrückt. Vgl. des weiteren BARR,R. (1989), S. 24, und WAND (1989), S. 549, zur inhaltlich eng verwandten Persistenzeigenschaft, die Objekten im Rahmen der objektorientierten Programmierung zukommt.

31) Dazu wird hier z.B. auch die räumliche Befindlichkeit von Objekten gerechnet.

32) Eine aktionsbedingte Veränderung der Objektstruktur läge nur dann vor, wenn Anzahl, Art oder Abhängigkeitsbeziehungen der Objekteigenschaften modifiziert würden. Dies wäre z.B. der Fall, wenn ein Objekt seine grundsätzliche Disposition, Träger einer bestimmten Eigenschaft zu sein, durch Ausführen einer Aktion einbüßte. Hierzu müßte z.B. ein Auftrag, der ehemals durch die Eigenschaft "Liefertermin" gekennzeichnet war, die Auszeichnung durch diese Eigenschaft grundsätzlich verlieren. Er müßte fortan als ein reales Objekt konzeptualisiert werden, daß prinzipiell nicht mehr fähig ist, eine Ausprägung der Eigenschaft "Liefertermin" auszuweisen.
Solche strukturellen Veränderungen der Konzeptualisierung von realen Objekten vermag sich der Verf. allerdings nicht vorzustellen - zumindest keine überzeugenden Rechtfertigungen für Veränderungen von Objektstrukturen: Wenn nach der Konzeptualisierung eines Realitätsausschnitts einmal eine bestimmte Objektstruktur eingeführt und plausibel belegt ist, so müßten bei einer Strukturveränderung die zuvor unterbreiteten Plausibilitätsargumente partiell widerrufen und durch andere ersetzt werden. Dies stellte aber die Argumentationsgrundlage der ursprünglich vorgelegten Plausibilitätsargumente nachträglich in Frage. Es drohte die Gefahr eines performativen Selbstwiderspruchs. Aufgrund dieser Schwierigkeiten, zu denen Modifizierungen der konzeptualisierten Objektstrukturen führen können, wird in dieser Arbeit von der Persistenzprämisse ausgegangen. Sie unterstellt, daß die Strukturen einmal konzeptualisierter realer Objekte im Zeitablauf erhalten bleiben, solange die Objekte selbst existieren.

Es muß jedoch zwei Mißverständnissen vorgebeugt werden. Erstens bedeutet die Persistenzprämisse, daß Veränderungen der *Ausprägungen* von Objekteigenschaften jederzeit geschehen können. Da oftmals Objekteigenschaften mit ihren Ausprägungen konfundiert werden, könnte der fehlerhafte Schluß gezogen werden, die Persistenzprämisse präsupponiere die Konstanz aller Eigenschaftsausprägungen. Dies ist jedoch überhaupt nicht der Fall, weil sich die Persistenzprämisse nicht auf Eigenschaftsausprägungen, sondern auf die involvierten Eigenschaften selbst bezieht. Zweitens steht die Persistenzprämisse unter einem Existenzvorbehalt. Sie schließt daher keineswegs aus, daß reale Aktionen Objekte und deren Eigenschaftsstrukturen erschaffen oder vernichten können. Dadurch ist es z.B. möglich, divergente, konvergente und umgruppierende Produktionsprozesse ohne Verletzung der Persistenzprämisse zu modellieren. Prozeßin- und -output können dabei verschiedene Objekt mit jeweils unterschiedlichen Objektstrukturen darstellen. Insbesondere bereitet es auf diese Weise keine Schwierigkeiten, den Zusammenbau komplexer Produkte (Objekte) in Montageprozessen zu erfassen. Darüber hinaus können auf diese Weise sogar die oben thematisierten Veränderungen in der Konzeptualisierung von Objektstrukturen abgedeckt werden, falls sie tatsächlich Relevanz erlangen sollten: Wenn eine Objektkonzeptualisierung strukturell variiert wird, gilt dies als Vernichtung des vormals konzeptualisierten und Erzeugung eines neuartig strukturierten Objekts.
Die Persistenzprämisse schließt also keineswegs aus, daß Objekte erschaffen oder vernichtet werden. Aber sie läßt solche ontischen Diskontinuitäten nur dann zu, wenn die Existenz des betroffenen Objekts tatsächlich beginnt bzw. endet. Dagegen wird verhindert, daß in einem Modell die Repräsentation eines Objekts zunächst verschwindet und später wieder auftaucht, obwohl das repräsentierte Objekt in der konzeptualisierten Realität durchgehend existiert. Solche vorübergehenden Objektausblendungen im Modell verletzen die ontische Verfassung des repräsentierten Realitätsausschnitts. Dennoch geschehen sie bei der Modellierung produktionswirtschaftlicher Sachverhalte des öfteren. Vgl. z.B. HOLT,A. (1975d), S. 158, zur scheinbaren, aber nur vorübergehenden "Vernichtung" eines Auftrags. Dieses Problem kann z.B. eintreten, wenn die Abwicklung des Auftrags eine Zeitlang nicht voranschreitet, weil etwa die zu bearbeitenden Werkstücke zwischengelagert sind. In dieser Zwischenzeit existiert das abstrakte Objekt "Auftrag" aber weiterhin. Die zwischengelagerten Werkstücke lassen sich dem Objekt "Auftrag" zuordnen. Wenn die Auftragsabwicklung durch neue Arbeitsgangausführungen wiederaufgenommen wird, entsteht kein neuer Auftrag.

33) Eine ähnliche Persistenzauffassung stellt WAND (1989), S. 549, auch als ontologisches Fundament der objektorientierten Systemgestaltung heraus: "Persistence ... enables changes of state without changing the essence of the thing." Dabei entsprechen "changes of state" den inhaltlich veränderten Objekteigenschaften und "essence of the thing" der invarianten Objektstruktur.

34) Diese Bedingung schließt diejenigen Sonderfälle aus, in denen Aktionen die Vernichtung alter oder die Erschaffung neuer Objekte bedeuten. Hierauf wurde in bereits in der voranstehenden Anmerkung aus der Perspektive des Existenzvorbehalts eingegangen.

35) Solche schwerwiegenden Integritätsverletzungen kommen in Modellierungsbeispielen der Netzliteratur durchaus vor. Vgl. etwa ABEL,D. (1990), S. 36f. u. 40. Es handelt sich um das Beispiel, das bereits in einer der voranstehenden Anmerkungen erläutert wurde. Bei ABEL wird verschwinden die Roboter "I" und "II" aus den Netzen Abb. 3.9, 3.10 u. 3.13, sobald sie ein Werkstück von einer der beiden Fertigungslinien ergriffen haben. Solange die Werkstücke gehandhabt werden, existiert im Netzmodell des Miniatur-Produktionssystems kein formales Objekt mehr, das den betroffenen Roboter repräsentieren würde. Nachdem die Werkstückhandhabung abgeschlossen ist, taucht der Roboter im Netzmodell plötzlich in der Gestalt einer roboterrepräsentierenden Marke wieder auf. Eine krassere Verletzung der Persistenz realer Objekte läßt sich nicht vorstellen. Ein ähnliches temporäres Objektverschwinden findet sich z.B. auch bei STAHLKNECHT (1991), Abb. A 6.22.1 auf S. 182 i.V.m. S. 148 u. 181f. (dort gehen Marken vorübergehend unter, die - je nach Betrachtungsweise - entweder Artikel oder aber Ausgabeschalter repräsentieren).

36) Auf eine andere - noch gravierendere - Integritätsverletzung wurde schon der voranstehenden Anmerkung hingewiesen.

37) Vgl. dazu die Ausführungen zur tendenziellen Gegenläufigkeit von Modellierungsgüte und -effizienz.

38) Ein typisches Beispiel für diese Vorgehensweise liefert ITTER (1989g), S. 3, Abb. 4. Dort werden bei der Modellierung einer Station, in der Container (Packkörbe) mit Werkstücken beladen werden, neben den Kennungen der jeweils betroffenen Objekte nur die beladungsrelevanten Objektgewichte als "Individuen"-Tupel erfaßt. Nachdem die Objekte die Beladungsstation verlassen haben, stehen Informationen über andere Objekteigenschaften, die vor der Beladungsstation bekannt gewesen sein mögen, bei *dieser* Modellierungsweise nicht mehr zur Verfügung.

39) Dabei wird unterstellt, daß dieses Wissen vor der hier betrachteten Transition noch als "Individuum" - d.h. Attribut - des werkstückabbildenden K-Tupels vorhanden war.

40) Dies geschieht durch die anonyme Variable "_" der Implementierungssprache Turbo-PROLOG. Sie erlaubt es, alle Markenattribute, die in der lokalen Nachbarschaft einer Transition für deren Schalten unerheblich sind, pauschal

zu kennzeichnen, ohne daß hierdurch die vorhandenen Informationen über die Attributausprägungen verloren gingen.

41) Sie wird daher auch als Markenontologie-Prämisse bezeichnet.

42) Der einfacheren Diktion halber wird auch von der Objektrepräsentation durch jeweils eine Marke gesprochen.

43) Dabei kann dasselbe reale Objekt im selben Zustand des Netzmodells auch durch mehrere Marken repräsentiert werden, die jeweils verschiedene Modellierungsperspektiven für dieses Objekt darstellen. Die Strukturen dieser Marken dürfen entsprechend zu den unterschiedlichen Modellierungsperspektiven voneinander abweichen. Das Strukturerhaltungspostulat gilt also nur jeweils innerhalb einer Modellierungsperspektive.

44) Jedes reale Objekt wird zwar durch eine Marke abgebildet. Doch es gilt nicht die Umkehrung. Denn Marken können herangezogen werden, um nicht-reale Sachverhalte abzubilden, wie z.B. logische Abhängigkeitsverhältnisse.

45) Eigenschafts- und Attributsbegriff werden synonym verwendet. Der einfacheren Diktion halber kann die korrekte Formulierung "Informationen über Eigenschaftsausprägungen" auch durch die verkürzte Ausdrucksweise "Informationen über Eigenschaften" ersetzt werden, sofern der Bezug auf Eigenschaftsausprägungen zumindest implizit gewahrt ist.

46) Es wird hierbei unterstellt, daß das modellierte reale Objekt mindestens eine modellierungsrelevante Eigenschaft besitzt. Eigenschaftslose reale Objekte können dagegen im Sinne des zweiten Informationsmodus hinsichtlich ihrer Existenz von Interesse sein.

47) Daneben spielen eigenschaftslose formale Objekte - die Kopien der Basismarke - auch für die Gestaltung logischer Sachverhalte eine Rolle.

48) Vgl. zu solchen Existenzinformationen DORN (1989), S. 67f.

49) Der einfacheren Diktion halber braucht fortan nur noch von "Marken" gesprochen zu werden. Gemeint sind hiermit jeweils Markenkopien oder Marken je nachdem, ob im Argumentationskontext formale Objekte bzw. Sorten thematisiert werden. Der Argumentationskontext kann sich auch - wie die anschließende Erläuterung des induktiven "Marken"-Aufbaus - simultan auf formale Objekte und Sorten erstrecken.

50) Durch das Instrument der multiplen Sortendefinition kann die zugrundeliegende Attributmarke auch die Zielsorte mehrerer verschiedenen Operationssymbole sein.

51) Es ist hier - wie auch sonst in dieser Arbeit - das logische "oder" gemeint, das die gleichzeitige Geltung beider Adjugatkomponenten einschließt. Daher kann eine Kompositmarke zugleich aus Attribut- und anderen Kompositmarken zusammengesetzt sein.

52) Durch die Möglichkeit, in den Argumenten von Restriktionsformeln unmittelbar auf Attributausprägungen Bezug zu nehmen, unterscheiden sich Restriktionsformeln deutlich von den voranstehenden Prädikatssymbolen und den daraus abgeleiteten Formeln. Restriktionsformeln stellen zwar ebenfalls prädikatenlogische Formeln dar, werden aber nicht der o.a. strukturellen Integritätsbedingungen unterworfen. Dies erleichtert später die Formulierung von transitionsspezifischen Schaltregeln.

5.1.1.3 Markenbezogene Spezifikationen

Markenbezogene algebraisch-prädikatenlogische MSIG-Spezifikationen[1)] werden auf der Grundlage von Markensignaturen definiert. Markensignaturen MSIG wurden aus prädikatenlogischen Signaturen PSIG dadurch abgeleitet, daß die Anforderungen einer Markenontologie Berücksichtigung fanden. Daraus werden nunmehr MSIG-Spezifikationen gewonnenen. Sie sorgen dafür, daß Netzmodelle die strukturellen Integritätsbedingungen für Objektmodellierungen erfüllen. Zugleich erweitern die MSIG-Spezifikationen das Konzept der algebraisch-prädikatenlogischen PSIG-Spezifikationen um den Aspekt einer operationalen Semantik. Die operationale Semantik wurde durch das allgemeine Übergangsschema und seine Konkretisierungen - den Übergangsoperationen - konstituiert. Sie werden nunmehr in die Definition der MSIG-Spezifikationen mittels der hinzugefügten Sektionen "ÜS" bzw. "trans" einbezogen.

Definition: markenbezogene algebraisch-prädikatenlogische Spezifikation

Eine markenbezogene algebraisch-prädikatenlogische MSIG-Spezifikation mit zugrundeliegender Markensignatur MSIG = (SO,OP,PRÄ) ist ein geordnetes 11-Tupel $SPEC_{MSIG}$ = (SO,OP,PRÄ; OBF,OPF,FAK_0;TTMF(VAF),VBM,RES;TR;ÜS), das folgendes Definitionsschema erfüllt:

<Spezifikationsname> =

sorts: $sort_1$ = $attribut_1$

...

$sort_Q$ = $attribut_Q$

$sort_{Q+1}$ = bas_marke

$sort_{Q+2}$ = att_$marke_1$

...

$sort_{Q+A+1}$ = att_$marke_A$

$sort_{Q+A+2}$ = str_$marke_1$

...

$sort_{Q+2A+B+1}$ = str_$marke_{A+B}$

$sort_{Q+2A+B+2}$ = sor_$marke_0$

...

$sort_{Q+3A+2B+2}$ = sor_$marke_{A+B}$

$SO = \{sort_i: i=1,...,I\}$ mit $I = Q+3A+2B+2$

Ops:

$Op_1 = Att_1$: $\rightarrow$ $attribut_{q(1)}$

...

$Op_{JQ} = Att_{JQ}$: $attribut_{q(JQ.1)} \ldots attribut_{q(JQ.KJQ)} \rightarrow attribut_{q(JQ.KJQ+1)}$

$Op_{JQ+1} = Marke_0$: $\rightarrow$ bas_marke

$Op_{JQ+2} = Marke_1$: $attribut_{q(JQ+2.1)} \ldots attribut_{q(JQ+2.KJQ+2)} \rightarrow att\_marke_{s(1)}$

...

$Op_{JQ+JA+1} = Marke_{JA}$: $attribut_{q(JQ+JA+1.1)} \ldots attribut_{q(JQ+JA+1.KJQ+JA+1)} \rightarrow att\_marke_{s(JA)}$

$Op_{JQ+JA+2} = Struk_1$: $att\_marke_1 \rightarrow str\_marke_1$

...

$Op_{JQ+A+JA+1} = Struk_A$: $att\_marke_A \rightarrow str\_marke_A$

$Op_{JQ+A+JA+2} = Struk_{A+1}$: $str\_marke_{s(A+1.1)} \ldots str\_marke_{s(A+1.KA+1)} \rightarrow str\_marke_{s(A+1.KA+1+1)}$

...

$Op_{JQ+A+JA+JB+1} = Struk_{A+JB}$: $str\_marke_{s(A+JB.1)} \ldots str\_marke_{s(A+JB.KA+JB)} \rightarrow str\_marke_{s(A+JB.KA+JB+1)}$

$Op_{JQ+A+JA+JB+2} = Sor_0$: $bas\_marke \rightarrow sor\_marke_0$

$Op_{JQ+A+JA+JB+3} = Sor_1$: $str\_marke_1 \rightarrow sor\_marke_1$

...

$Op_{JQ+2A+B+JA+JB+2} = Sor_{A+B}$: $str\_marke_{A+B} \rightarrow sor\_marke_{A+B}$

$OP = \{Op_j : j = 1,\ldots,J\}$ mit $J = J_Q + J_A + J_B + 2 + 2A + B$

Präs:

$Prä_1(sor\_marke_{s(1.1)} \ldots sor\_marke_{s(1.K1)})$

...

$Prä_U(sor\_marke_{s(U.1)} \ldots sor\_marke_{s(U.KU)})$

$PRÄ = \{Prä_u : u=1,\ldots,U\}$

OBs:

$OB_1 = OB_{att.1}$

...

$OB_Q = OB_{att.Q}$

$OB_{Q+1} = OB_{bas\_marke} = SYMBOL$

$OB_{Q+1+a} = OB_{att\_marke} = SYMBOL$ für $a \in \{1,\ldots,A\}$

$OB_{Q+1+A+b} = OB_{str\_marke} = SYMBOL$ für $b \in \{1,\ldots,A+B\}$

$OB_{Q+1+2A+B+s} = OB_{sor\_marke} = SYMBOL$ für $s \in \{0,\ldots,A+B\}$

$OBF = (OB_i : i=1,\ldots,I)$

ops:	$op_1 = att_1$:	$\rightarrow OB_{att.q(1)}$
		$att_1() = at_1$
	...	
	$op_{JQ} = att_{JQ}$:	$OB_{att.q(JQ.1)} \times \ldots \times OB_{att.q(JQ.KJQ)} \rightarrow OB_{att.q(JQ.KJQ+1)}$
		$(at_{JQ.1},\ldots,at_{JQ.KJQ}) \rightarrow att_{JQ}(at_{JQ.1},\ldots,at_{JQ.KJQ}) = at_{JQ.KJQ+1}$
	$op_{JQ+1} = marke_0$:	$\rightarrow OB_{bas\_marke}$
		$marke_0() = \varnothing = m_0$
	$op_{JQ+2} = marke_1$:	$OB_{att.q(JQ+2.1)} \times \ldots \times OB_{att.q(JQ+2.KJQ+2)} \rightarrow OB_{att\_marke}$
		$(at_{JQ+2.1},\ldots,at_{JQ+2.KJQ+2})$
		$\rightarrow marke_1(at_{JQ+2.1},\ldots,at_{JQ+2.KJQ+2}) = m_{s(1)}$
	...	
	$op_{JQ+JA+1}$	$OB_{att.q(JQ+JA+1.1)} \times \ldots \times OB_{att.q(JQ+JA+1.KJQ+JA+1)} \rightarrow OB_{att\_marke}$
	$= marke_{JA}$:	$(at_{JQ+JA+1.1},\ldots,at_{JQ+JA+1.KJQ+JA+1})$
		$\rightarrow marke_A(at_{JQ+JA+1.1},\ldots,at_{JQ+JA+1.KJQ+JA+1}) = m_{s(JA)}$
	$op_{JQ+JA+2}$	$OB_{att\text{-}marke} \rightarrow OB_{str\_marke}$
	$= struk_1$:	$m_1 \rightarrow struk_1(m_1) = m_1$
	...	
	$op_{JQ+A+JA+1}$	$OB_{att\text{-}marke} \rightarrow OB_{str\_marke}$
	$= struk_A$:	$m_A \rightarrow struk_A(m_A) = m_A$
	$op_{JQ+A+JA+2}$	$OB_{str\_marke} \times \ldots \times OB_{str\_marke} \rightarrow OB_{str\_marke}$
	$= struk_{A+1}$:	$(m_{s(A+1.1)},\ldots,m_{s(A+1.KA+1)}) \rightarrow \ldots$
		$struk_{A+1}(m_{s(A+1.1)},\ldots,m_{s(A+1.KA+1)}) = m_{s(A+1.KA+1+1)}$
	...	
	$op_{JQ+A+JA+JB+1}$	$OB_{str\_marke} \times \ldots \times OB_{str\_marke} \rightarrow OB_{str\_marke}$
	$= struk_{A+JB}$:	$(m_{s(A+JB.1)},\ldots,m_{s(A+JB.KA+JB)}) \rightarrow \ldots$
		$struk_{A+JB}(m_{s(A+JB.1)},\ldots,m_{s(A+JB.KA+JB)}) = m_{s(A+JB.Ka+JB+1)}$
	$op_{JQ+A+JA+JB+2}$	$OB_{bas\_marke} \rightarrow OB_{sor\_marke}$
	$= sor_0$:	$m_0 \rightarrow sor_0(m_0) = m_0$
	$op_{JQ+A+JA+JB+3}$	$OB_{str\_marke} \rightarrow OB_{sor\_marke}$
	$= sor_1$:	$m_1 \rightarrow sor_1(m_1) = m_1$
	...	
	$op_{JQ+2A+B+JA+JB+2}$	$OB_{str\_marke} \rightarrow OB_{sor\_marke}$
	$= sor_{A+B}$:	$m_{A+B} \rightarrow sor_{A+B}(m_{A+B}) = m_{A+B}$

$OPF = (op_j\text{: } j=1,\ldots,J)$ mit $J = J_Q + J_A + J_B + 2 + 2A + B$

<u>präs</u>: $prä_1$: $OB_{sor\text{-}marke} \times \ldots \times OB_{sor\_marke}$

$fakt_0(mu_{1.0.1}, prä_1(m_{s(1.1.1)}, \ldots, m_{s(1.K1.1)}))$

...

$fakt_0(mu_{1.0.D1}, prä_1(m_{s(1.1.D1)}, \ldots, m_{s(1.K1.D1)}))$

$FAK_{1.0} = \{fakt_0(mu_{1.0.d}, prä_1(m_{s(1.1.d)}, \ldots, m_{s(1.K1.d)})): d = 1, \ldots, D_1\}$

...

$prä_U$: $OB_{sor\text{-}marke} \times \ldots \times OB_{sor\_marke}$

$fakt_0(mu_{U.0.1}, prä_U(m_{s(U.1.1)}, \ldots, m_{s(U.KU.1)}))$

...

$fakt_0(mu_{U.0.DU}, prä_U(m_{s(U.1.DU)}, \ldots, m_{s(U.KU.DU)}))$

$FAK_{U.0} = \{fakt_0(mu_{U.0.d}, prä_U(m_{s(U.1.d)}, \ldots, m_{s(U.KU.d)})): d = 1, \ldots, D_U\}$

$FAK_0 = (FAK_{u.0}: u = 1, \ldots, U)$

<u>Vars</u>: $VA_1 = \{X_{1.1}, \ldots, X_{1.V1}\}$

...

$VA_I = \{X_{I.1}, \ldots, X_{I.VI}\}$

$VAF = (VA_i: i = 1, \ldots, I)$

<u>terms</u>: $KB_1 = \{te: \exists(op_j \in OPF): op_j: \rightarrow OB_1 \wedge op_j() = te\}$

$TTM_1(VA_1)$ minimal mit:

$$
\begin{aligned}
& KB_1 = \emptyset \rightarrow OB_1 \subseteq TTM_1(VA_1) \\
\wedge\ & KB_1 \neq \emptyset \rightarrow KB_1 \subseteq TTM_1(VA_1) \\
\wedge\ & VA_1 \subseteq TTM_1(VA_1) \\
\wedge\ & (\forall(op_j \in OPF): \ldots \\
& \quad (op_j: OB_{i(j.1)} \times \ldots \times OB_{i(j.Kj)} \rightarrow OB_{i(j.Kj+1)} \\
& \quad \wedge (\forall(k \in \{1, \ldots, K_j\}): te_k \in TTM_{i(j.k)}(VA_{i(j.k)})) \\
& \quad \wedge i(j.K_j+1) = 1 \wedge op_j(te_1, \ldots, te_{Kj}) = te_{Kj+1}) \\
& \rightarrow te_{Kj+1} \in TTM_1(VA_1))
\end{aligned}
$$

...

$KB_I = \{te: \exists(op_j \in OPF): op_j: \rightarrow OB_I \wedge op_j() = te\}$

$TTM_I(VA_I)$ minimal mit:

$$
\begin{aligned}
& KB_I = \emptyset \rightarrow OB_I \subseteq TTM_I(VA_I) \\
\wedge\ & KB_I \neq \emptyset \rightarrow KB_I \subseteq TTM_I(VA_I) \\
\wedge\ & VA_I \subseteq TTM_I(VA_I) \\
\wedge\ & (\forall(op_j \in OPF): \ldots \\
& \quad (op_j: OB_{i(j.1)} \times \ldots \times OB_{i(j.Kj)} \rightarrow OB_{i(j.Kj+1)} \\
& \quad \wedge (\forall(k \in \{1, \ldots, K_j\}): te_k \in TTM_{i(j.k)}(VA_{i(j.k)})) \\
& \quad \wedge i(j.K_j+1) = I \wedge op_j(te_1, \ldots, te_{Kj}) = te_{Kj+1}) \\
& \rightarrow te_{Kj+1} \in TTM_I(VA_I))
\end{aligned}
$$

$TTMF(VAF)=(TTM_i(VA_i):i=1,...,I)$

$VBM =\{vb_c: c=1,...,C \wedge C\in \mathcal{N}_+ \wedge (\forall(c\in\{1,...C\}: K_c\in \mathcal{N}_+ ...$
$\quad vb_c: TTM_{i(1)}(VA_{i(1)}) \times ... \times TTM_{i(Kc)}(VA_{i(Kc)}) \rightarrow OB_{i(1)} \times ... \times OB_{i(Kc)})\}$

res: a) entweder für Z=0: $RES = FOR_{standard}$

b) oder aber für $Z\in \mathcal{N}_+$:

$for_1(te_1,...,te_{K1})$

...

$for_z(te_1,...,te_{Kz})$

mit:

$\forall(z\in\{1,...,Z\})\ \exists(K_z\in \mathcal{N}_+)\ \forall(k\in\{1,...,K_z\}): te_k\in TTM_{i(k)}(VA_{i(k)})$

$RES = \{for_z(te_1,...,te_{Kz}): z=1,...,Z\} \cup FOR_{standard}$

trans: $tr_1 = ...$

$(VB(tr_1) = \{Prä_u(sor\_marke_{s(u.1)},...,sor\_marke_{s(u.Ku)}): u\in IVB_1\},$
$MTAV_1 = \{MTAV_{u.1}: u\in IVB_1\},$
$IB(tr_1) = \{Prä_u(sor\_marke_{s(u.1)},...,sor\_marke_{s(u.Ku)}): u\in IIB_1\},$
$MTAI_1 = \{MTAI_{u.1}: u\in IIB_1\},$
$NB(tr_1) = \{Prä_u(sor\_marke_{s(u.1)},...,sor\_marke_{s(u.Ku)}): u\in INB_1\},$
$MTAN_1 = \{MTAN_{u.1}: u\in INB_1\},$
$RES_1 \in \{\emptyset, \{for_{z(1.h)}: \underline{\exists}(H_1\in \mathcal{N}_+)\ \forall(h\in\{1,...,H_1\}): for_{z(1.h)}\in RES\}\})$

...

$tr_V = ...$

$(VB(tr_V) = \{Prä_u(sor\_marke_{s(u.1)},...,sor\_marke_{s(u.Ku)}): u\in IVB_V\},$
$MTAV_V = \{MTAV_{u.V}: u\in IVB_V\},$
$IB(tr_V) = \{Prä_u(sor\_marke_{s(u.1)},...,sor\_marke_{s(u.Ku)}): u\in IIB_V\},$
$MTAI_V = \{MTAI_{u.V}: u\in IIB_V\},$
$NB(tr_V) = \{Prä_u(sor\_marke_{s(u.1)},...,sor\_marke_{s(u.Ku)}): u\in INB_V\},$
$MTAN_V = \{MTAN_{u.V}: u\in INB_V\},$
$RES_V \in \{\emptyset, \{for_{z(V.h)}: \underline{\exists}(H_V\in \mathcal{N}_+)\ \forall(h\in\{1,...,H_V\}): for_{z(V.h)}\in RES\}\})$

$TR = \{tr_v: v=1,...,V\}$

ÜS: <Übergangsschema-Deklaration>

Erläuterungen und Ergänzungen zur Definition einer markenbezogenen algebraisch-prädikatenlogischen Spezifikation:

a) Die Sorten "bas_marke", "att_marke", "str_marke" und "sor_marke" präzisieren in formalsprachlicher Weise die natürlichsprachlich eingeführten Basis- und Attributmarken sowie strukturierten und sortierten Marken. Sie werden daher als Markensorten bezeichnet. Jede Markensorte repräsentiert in einer MSIG-Spezifikation genau eine Marke. Daher wird die Markensorte auch als Name der repräsentierten Marke verwendet[2)].

b) Die Sorten "attribut_q" mit $q \in \{1,...,Q\}$ und $Q \in \mathcal{N}_+$ werden für die Definition von Objekteigenschaften (Attributen) benutzt. Sie heißen Attributsorten. Jedes Attribut wird in einer MSIG-Spezifikation durch genau eine Attributsorte repräsentiert. Jeder Attributsorte entspricht genau ein Attribut. Wegen dieser bijektiven Zuordnung können Attribute und Attributsorten miteinander identifiziert werden[3)]. Attributsorten werden daher auch kurz als Attribute bezeichnet.

c) $OB_{att.q}$ ist die Menge aller Attributausprägungen "at", die für eine Attributsorte "attribut_q" mit $q \in \{1,...Q\}$ grundsätzlich in Betracht kommen. Sie werden als mögliche Ausprägungen des Attributs "attribut_q" bezeichnet. Die Menge $OB_{att.q}$ aller möglichen Attributausprägungen heißt auch Definitions- oder Wertebereich des zugrundeliegenden Attributs. Die Attributkonstitution kann sowohl einfach als auch komplex sein.

d) Bei einer einfachen Attributkonstitution stellt das Attribut eine originär definierte Attributsorte "attribut_q" dar[4)]. Die möglichen Attributausprägungen "at" sind dann originär definierte formale Konstrukte, die sich nicht mehr in andere formale Konstrukte zerlegen lassen. Es handelt sich um atomare formale Objekte[5)] aus dem Definitionsbereich $OB_{att.q}$ der Attributsorte "attribut_q". Die zulässigen Attributausprägungen[6)] werden entweder als Bilder von nullstelligen Operationen op_j mit $op_j{:} \rightarrow OB_{att.q}$ eingeführt. Ihre Gesamtheit bildet dann eine echte Teilmenge des Definitionsbereichs $OB_{att.q}$. Oder aber es sind überhaupt keine nullstelligen Operationen op_j mit $op_j{:} \rightarrow OB_{att.q}$ in der Operationenfamilie OPF definiert. In diesem Fall stimmt die Menge aller zulässigen Attributausprägungen mit dem Definitionsbereich $OB_{att.q}$ der Attributsorte "attribut_q" überein.

e) Bei einer komplexen Attributkonstitution ist das Attribut in derivativer Weise als Zielsorte "$\text{attribut}_{q(j.Kj+1)}$" eines Operationssymbols "Att_j" definiert[7)]. Dann wird diese derivative Attributsorte durch die Attributsorten "$\text{attribut}_{q(j.k)}$" aus dem Argument ihres Operationssymbols "Att_j" mit $k \in \{1,...,K_j\}$ aufgebaut. Bei den möglichen Ausprägungen des Attributs handelt es sich um zusammengesetzte formale Objekte aus dem Definitionsbereich $OB_{att.q(j.Kj+1)}$ der Zielsorte "$\text{attribut}_{q(j.Kj+1)}$". Für diesen Definitionsbereich wird - falls keine andere Festlegung erfolgt - stets die globale Objektmenge "SYMBOL" gewählt. Die zulässigen[8)] Attributausprägungen sind mittelbar festgelegt: Sie umfassen alle Kombinationen aus zulässigen Ausprägungen jener Attribute, die als Attributsorten "$\text{attribut}_{q(j.k)}$" im Argument des Operationssymbols "Att_j" vorkommen.

f) Dieselbe Attributsorte darf sowohl originär als auch derivativ definiert sein[9)]. Ebenso kann eine Attributsorte auf unterschiedliche Weisen derivativ definiert werden. Dann ist sie die Zielsorte von mehreren verschiedenen Operationssymbolen "Att_j"[10)]. J_Q ist die Anzahl aller derivativen Attributsortendefinitionen, welche die MSIG-Spezifikation mit $J_Q \in \mathcal{N}_0$ enthält. Im Grenzfall $J_Q=0$ ist keine der Q Attributsorten in derivativer Weise definiert. Durch die unbeschränkte Möglichkeit, Attributsorten und Attributausprägungen in beliebig vielen Stufen aus anderen Attributsorten bzw. -ausprägungen zusammensetzen, wird der Strukturierungsreichtum des algebraischen Signaturkonzepts voll ausgeschöpft.

g) Das 0-stellige Operationssymbol "Marke_0" symbolisiert die Erzeugungsoperation "marke_0" für alle Kopien m_0 der Basismarke. Die Basismarke mit der Sorte "bas_marke" ist die Anwendung des Erzeugungs-Operationssymbols "Marke_0" auf das leere Argument "()": bas_marke= $\mathrm{Marke}_0()$. Die Basismarke ist in jeder Spezifikation $\mathrm{SPEC}_{\mathrm{MSIG}}$ definiert.

h) Jede Kopie m_0 der Basismarke ist eine Anwendung der Erzeugungsoperation "marke_0" auf das leere Argument "()": $m_0=\mathrm{marke}_0()$. Als dritte Notation für Kopien der Basismarke wird auch das Symbol "$\varnothing$" zugelassen. Die drei markenspezifischen Konstanten "m_0", "$\mathrm{marke}_0()$" und "$\varnothing$" werden als identisch behandelt: $m_0=\mathrm{marke}_0()=\varnothing$.

i) Die Basismarke und ihre Kopien besitzen keine innere Struktur. Die Basismarke heißt auch unstrukturierte, strukturlose oder farblose Marke.

j) Alle Kopien der Basismarke sind notwendig identisch. Da sie keine innere Struktur aufweisen, können sie sich in keinem Aspekt unterscheiden.

k) Die K_j-stelligen Operationssymbole "Marke_j" mit den Zielsorten $\mathrm{att\_marke}_{s(j)}$, $j\in\{1,...,J_A\}$ und $K_j\in \mathcal{N}_+$ symbolisieren die Erzeugungsoperationen "marke_j" für alle Kopien m_a von Attributmarken der Sorten $\mathrm{att\_marke}_a$ mit $a=s(j)$ und $a\in\{1,...,A\}$. A ist die Anzahl von Attributmarken, die in der MSIG-Spezifikation mit $A\in \mathcal{N}_0$ definiert sind. Für A=0 ist überhaupt keine Attributmarke definiert. Als Standardfall wird jedoch fortan $A\in \mathcal{N}_+$ vorausgesetzt. Für jede definierte Attributmarke existiert mindestens ein Operationssymbol "Marke_j"[11]. Daher gilt immer $J_A\geq A$. Dies schließt die Möglichkeit ein, daß dieselbe Attributmarke durch mehrere verschiedene Operationssymbole "Marke_j" erzeugt wird[12]. Die Attributmarke ist daher entweder die Zielsorte $\mathrm{att\_marke}_a$ von genau einem Operationssymbol "Marke_j" mit $a=s(j)$. Oder die Attributmarke ist die identische Zielsorte $\mathrm{att\_marke}_{s(j(1))}=...=\mathrm{att\_marke}_{s(j(K_a))}$ von mehreren verschiedenen Operationssymbolen $\mathrm{Marke}_{j(1)},...,\mathrm{Marke}_{j(K_a)}$ mit $K_a\geq 2$ und $a=s(j(1))=...=s(j(K_a))$. Im letzten Fall liegt eine multiple Attributmarkendefinition vor.

l) Das Argument "$\mathrm{attribut}_{q(j.1)}$ ... $\mathrm{attribut}_{q(j.Kj)}$" des Operationssymbols "Marke_j" für die Erzeugung einer Attributmarke der Sorte $\mathrm{att\_marke}_a$ mit $a=s(j)$ heißt die Attributstruktur der Marke. Sie legt Anzahl, Art und Anordnung aller Markenattribute durch die Attributsorten $\mathrm{attribut}_{q(j.k)}$ mit $k\in\{1,...,K_j\}$ fest. Falls eine multiple Attributmarkendefinition vorliegt, sind derselben Attributmarke durch verschiedene Operationssymbole "Marke_j" mit differierenden Argumenten mehrere unterschiedliche Attributstrukturen zugeordnet.

m) Jede Kopie m_a einer Attributmarke mit $a\in\{1,...,A\}$, die als Zielsorte $\mathrm{att\_marke}_{s(j)}$ eines Erzeugungs-Operationssymbols "Marke_j" mit $j\in\{1,...,J_A\}$ und $a=s(j)$ definiert ist, stellt ein Bild der Erzeugungsoperation "marke_j" für ein Argument $(at_{j.1},...,at_{j.Kj})$ dar:

$$m_a = m_{s(j)} = \mathrm{marke}_j(at_{j.1},...,at_{j.Kj})$$

n) Kopien derselben Attributmarke heißen identisch (verschieden) genau dann, wenn die Erzeugungsoperation "marke_j" auf identische (verschiedene) Argumente $(at_{j.1},...,at_{j.Kj})$ angewandt worden ist. Im Fall einer multiplen Attributmarkendefinition sind die Kopien, die Bilder von unterschiedlichen Erzeugungsoperationen darstellen, immer verschieden.

o) Für jede Attributmarke können beliebig, aber höchstens endlich viele identische oder verschiedene Kopien existieren. Alle Kopien m_a derselben Attributmarke mit der Sorten $\mathrm{att\_marke}_a$ werden als artgleiche Markenkopien oder als Kopien der Markenart "a" bezeichnet. Verschiedene Kopien derselben Attributmarke können durch einen differenzierenden Index "d" unterschieden werden:

$$m_{a.d} = m_{s(j).d} = marke_j(at_{j.1.d},...,at_{j.Kj.d})$$

p) Das Argument $(at_{j.1},...,at_{j.Kj})$ der Erzeugungsoperation $marke_j$ für die Attributmarkenkopie m_a mit $m_a = m_{s(j)} = marke_j(at_1,...,at_{Kj})$ ist die Attributstruktur der Markenkopie m_a. Die Attributstruktur determiniert für jedes Attribut $attribut_{q(j.k)}$ der Attributmarke mit der Sorte att_marke_a eine Attributausprägung $at_{j.k}$ mit $at_{j.k} \in OB_{q(j.k)}$ für alle $k \in \{1,...,K_j\}$.

q) Die Attributstrukturen von Attributmarken und deren Kopien sind Tupel aus Attributsorten bzw. Attributausprägungen. Die Tupel können sowohl flach als auch tief strukturiert sein. Im ersten Fall stellen alle Komponenten des Tupels originär definierte Attributsorten bzw. atomare formale Objekte für die Attributausprägungen dar. Im zweiten Fall handelt es sich bei mindestens einer Tupelkomponente um eine derivativ definierte Attributsorte bzw. um ein zusammengesetztes formales Objekt als Attributausprägung. Dann wird auch von einer hierarchisch verschachtelten Attributstruktur gesprochen.

r) Die K_j-stelligen Operationssymbole "$Struk_j$" mit $j \in \{1,...,A+J_B\}$ und $K_j \in \mathcal{N}_+$ symbolisieren die Strukturierungsoperationen "$struk_j$" für alle Kopien m_b von strukturierten Marken mit $b \in \{1,...,A+B\}$. A+B ist die Anzahl von strukturierten Marken, die in der MSIG-Spezifikation mit $(A+B) \in \mathcal{N}_0$ definiert sind. Die Anzahl A unterschiedlicher Attributmarken wurde bereits erläutert. B ist die Anzahl unterschiedlicher Kompositmarken mit $B \in \mathcal{N}_0$. Für jede Attributmarke existiert genau ein Strukturierungs-Operationssymbol und für jede Kompositmarke mindestens ein Strukturierungs-Operationssymbol. Für die Strukturierung der Kompositmarken gilt daher $J_B \geq B$[13)].

s) Für B=0 existieren keine Kompositmarken. Dann fallen die strukturierten Marken mit den Attributmarken zusammen. Falls wegen A=0 auch keine Attributmarken definiert sind, existieren mit S=A+B=0+0=0 überhaupt keine strukturierten Marken. Dann sind nur die Basismarke mit $bas\_marke = Marke_0()$ und ihre Kopien $m_0 = marke_0()$ definiert. In diesem Grenzfall liegt eine algebraisch-prädikatenlogische Spezifikation für die Marke(nkopien) von Stelle/Transition-Netzen vor. Als Standardfall wurde jedoch bereits $A \in \mathcal{N}_+$ vorausgesetzt, so daß in der Regel $(A+B) \in \mathcal{N}_+$ gilt und mindestens eine strukturierte Marke definiert ist.

t) Bei den 1-stelligen Operationssymbolen "$Struk_j$" mit den Zielsorten str_marke_j und $j \in \{1,...,A\}$ handelt es sich jeweils um das Symbol der identischen Abbildung. Hierdurch wird jede Attributmarke att_marke_a als eine atomare strukturierte Marke str_marke_j mit j=a ausgewiesen. Andere atomare strukturierte Marken als die Attributmarken gibt es nicht.

u) Die K_j-stelligen Operationssymbole "$Struk_j$" mit den Zielsorten $str\_marke_{s(j.Kj+1)}$, $j \in \{A+1,...,A+J_B\}$ und $K_j \in \mathcal{N}_+$ symbolisieren jeweils eine Kompositionsoperation. Hierdurch werden strukturierte Marken aus anderen strukturierten Marken zusammengesetzt. In die Markenzusammensetzung können sowohl Attributmarken als auch andere, bereits eingeführte Kompositmarken eingehen. Das Zusammensetzungsergebnis ist eine zusammengesetzte strukturierte Marke der Sorte str_marke_{A+b} mit $b \in \{1,...,B\}$ und $A+b = s(j.K_j+1)$. Diese Marke wird als Kompositmarke oder komplexe Marke bezeichnet. Die Kompositmarke str_marke_{A+b} ist entweder die Zielsorte $str\_marke_{s(j.Kj+1)}$ von genau einem Operationssymbol "$Struk_j$" mit $A+b = s(j.K_j+1)$. Oder die Kompositmarke ist die identische Zielsorte $str\_marke_{s(j(1).Kj(1)+1)} = ... = str\_marke_{s(j(Kb).Kj(Kb)+1)}$ von mehreren verschiedenen Operationssymbolen $Struk_{j(1)},...,Struk_{j(Kb)}$ mit $K_b \geq 2$ und $A+b = s(j(1).K_{j(1)}+1) = ... = s(j(K_b).K_{j(Kb)}+1))$. Im letzten Fall liegt eine multiple Kompositmarkendefinition vor.

v) Das Argument "$str\_marke_{s(j.1)}$ ... $str\_marke_{s(j.Kj)}$" des Operationssymbols "$Struk_j$" mit der Zielsorte $str\_marke_{s(j.Kj+1)}$ heißt die Zusammensetzungsstruktur der Kompositmarke der Sorte str_marke_{A+b} mit $b \in \{1,...,B\}$ und $A+b = s(j.K_j+1)$. Diese Zusammensetzungsstruktur legt durch die Markensorten $str\_marke_{s(j.k)}$ mit $k \in \{1,...,K_j\}$ Anzahl, Art und Anordnung aller strukturierten Marken fest, die am Aufbau der Kompositmarke str_marke_{A+b} beteiligt sind. Bei einer multiplen Kompositmarkendefinition werden derselben Kompositmarke durch verschiedene Operationssymbole mit differierenden Argumenten mehrere unterschiedliche Zusammensetzungsstrukturen zugeordnet.

w) Jede Kopie m_a einer atomaren strukturierten Marke der Sorte str_marke_a mit $a \in \{1,...,A\}$ ist mit a=j ein Bild m_j der identischen Strukturierungsoperation "$struk_j$" für das Argument m_j. Da die atomare strukturierte Marke eine Attributmarke ist, kann die Markenkopie m_j ihrerseits als das Bild $m_{s(h)}$ einer Erzeugungsoperation "$marke_h$" mit $j=s(h)$ aufgefaßt werden. Daher gilt:

$$m_a = struk_j(m_j) = marke_h(at_{h.1},...,at_{h.Kh}) = m_{s(h)} \quad \text{für } a=j=s(h)$$

x) Jede Kopie m_{A+b} einer zusammengesetzten strukturierten Marke der Sorte str_marke_{A+b} mit $b \in \{1,...,B\}$, die als Zielsorte $str\_marke_{s(j.Kj+1)}$ eines Strukturierungs-Operationssymbols "$Struk_j$" mit $s(j.K_j+1) = A+b$ definiert ist, stellt ein Bild $m_{s(j.Kj+1)}$ der Strukturierungsoperation "$struk_j$" für ein Argument $(m_{s(j.i)},...,m_{s(j.Kj)})$ dar:

$$m_{A+b} = m_{s(j.Kj+1)} = struk_j(m_{s(j.i)},...,m_{s(j.Kj)}) \quad \text{für } A+b=s(j.K_j+1)$$

y) Kopien derselben Kompositmarke heißen identisch (verschieden) genau dann, wenn die Strukturierungsoperation "$struk_j$" auf identische (verschiedene) Argumente $(m_{s(j.i)},...,m_{s(j.Kj)})$ angewandt worden ist. Im Fall einer multiplen Kompositmarkendefinition sind die Kopien, die Bilder von unterschiedlichen Strukturierungsoperationen darstellen, immer verschieden.

z) Für jede Kompositmarke können beliebig, aber höchstens endlich viele identische oder verschiedene Kopien existieren. Alle Kopien derselben Kompositmarke str_marke_j werden als artgleiche Markenkopien oder als Kopien der Markenart "A+b" bezeichnet. Verschiedene Kopien derselben Kompositmarke können durch einen differenzierenden Index "d" unterschieden werden:

$$m_{A+b} = m_{s(j.Kj+1).d} = struk_j(m_{s(j.i).d},...,m_{s(j.Kj).d})$$

A) Das Argument $(m_{s(j.i)},...,m_{s(j.Kj)})$ der Strukturierungsoperation "$struk_j$" für die Kopie $m_{A+b} = m_{s(j.Kj+1)} = struk_j(m_{s(j.i)},...,m_{s(j.Kj)})$ einer Kompositmarke ist die Zusammensetzungsstruktur der Markenkopie. Sie gibt an, auf welche Weise die Markenkopie aus den Kopien anderer strukturierter Marken aufgebaut ist.

B) Die Zusammensetzungsstrukturen von Kompositmarken können in hierarchischer Weise beliebig tief verschachtelt werden. Dazu werden die Markenkopien $m_{s(j.k)}$, die mit $k \in \{1,...,K_j\}$ an der Zusammensetzungsstruktur einer Markenkopie $M_{A+b} = m_{s(j.Kj+1)}$ teilnehmen, ihrerseits als Bilder $m_{s(h.Kh+1)}$ von Strukturierungsoperationen "$struk_h$" mit $j.k = s(h.K_h+1)$ ausgedrückt:

$$m_{s(j.k)} = m_{s(h.Kh+1)} = struk_h(m_{s(h.1)},...,m_{s(h.Kh)})$$

C) Die Basismarke und alle Attributmarken stellen die kleinsten selbständig existenzfähigen Objekte dar. Attributmarken bilden die unterste Hierarchiestufe der Zusammensetzungsstrukturen von Kompositmarken. Kompositmarken, deren Kopien nur aus Attributmarkenkopien zusammengesetzt sind, heißen daher Kompositmarken erster Ordnung[14)]. Bei allen anderen Kompositmarken, deren Kopien jeweils mindestens eine andere, bereits eingeführte Kompositmarkenkopie umfassen, handelt es sich um Kompositmarkenkopien höherer Ordnung[15)].

D) Der Strukturierungsreichtum, den das algebraische Universum im Signaturkonzept eröffnet, wird sowohl durch die Attributstrukturen der Argumente von Attributmarken als auch durch die Zusammensetzungsstrukturen der Argumente von Kompositmarken jeweils vollständig ausgeschöpft. In beiden Fällen ist es möglich, die Argumentstrukturen in beliebig komplexer Weise hierarchisch aus jeweils einfacheren Komponenten aufzubauen. Die einfachsten, nicht mehr weiter zerlegbaren Komponenten stellen immer originär definierte Attributsorten und die atomaren formalen Objekte ihrer zugehörigen Attributausprägungen dar.

E) Die Attribut- und die Zusammensetzungsstrukturen von Attribut- bzw. Kompositmarken (oder deren Kopien) werden unter den Oberbegriff der Markenstruktur einer (Kopie einer) strukturierten Marke subsumiert.

F) Die 1-stelligen Operationssymbole "Sor_s" besitzen lediglich technischen Charakter. Sie dienen dazu, alle zuvor eingeführten Marken als sortierte Marken der Sorten "sor_marke_s" mit $s \in \{0,1,...,S\}$, $S=A+B$ und $S \in \mathcal{N}_0$ zu deklarieren. Die zugehörigen Operationen "sor_s" stellen jeweils die identische Abbildung dar. Sie unterscheiden sich aber voneinander durch ihre verschiedenen Argument- und Zielsorten.

G) Aufgrund der Identitäts-Operationssymbole "Sor_s" kann jede sortierte Marke entweder auf die eine Basismarke "bas_marke" oder auf eine strukturierte Marke reduziert werden. Als strukturierte Marke ist sie entweder eine Attributmarke "att_marke_a" mit $a \in \{1,...,A\}$ oder aber eine Kompositmarke "str_marke_{A+b}" mit $b \in \{1,...,B\}$.

H) Für sortierte Marken und deren Kopien gelten alle voranstehend definierten Konstrukte wie für Basis-, Attribut- und Kompositmarken. Diese Konstrukte werden hier nicht wiederholt, sondern als vereinbart vorausgesetzt.

I) Eine Markenart "s" mit $s \in \{0,1,...,S\}$, $S=A+B$ und $S \in \mathcal{N}_0$ ist der gemeinsame Index aller Markenkopien m_s, die Kopien einer Marke der Sorte "sor_marke_s" sind. Markenarten "s" und Markensorten "sor_marke_s" sind daher auf eineindeutige Weise einander zugeordnet. Mit $MAM=\{s\text{: } s=0,1,...,S\}$ wird die Markenartenmenge der Spezifikation MSIG bezeichnet.

J) Folgende markenbezogenen Ausdrücke entsprechen einander eineindeutig:

- "Marke" ist ein natürlichsprachlicher Ausdruck.
- Eine Markensorte "sor_marke_s" ist der formalsprachlicher Ausdruck, der genau einen natürlichsprachlichen Ausdruck "Marke" präzisiert.
- Eine Markenart "s" ist der identifizierende Index der Markensorte "sor_marke_s".

K) Die Kopien m_s derselben Markenart "s" können, müssen aber nicht identisch sein[16)]. Verschiedene Kopien derselben Markenart "s" können durch den differenzierenden Index "d" mit $d \in \{1,...,D_s\}$ und $D_s \in \mathcal{N}_+$ voneinander unterschieden werden.

L) $SMM=\{m_{s,d}\text{: } s=0,1,...,S \;\; d=1,...,D_s\}$ mit $S \in \mathcal{N}_0$ ist die Menge aller verschiedenen Kopien m_s von sortierten Marken, die durch die Spezifikation $SPEC_{MSIG}$ als Markensorten sor_marke_s definiert sind. Die Menge SMM wird auch als sortierte Markenmenge oder Markenuniversum bezeichnet.

M) Der sortierten Markenmenge SMM ist per constructionem eine Artenstruktur überlagert. Sie wird durch die beteiligten Markenarten "s" konstituiert:

- Die Kopien $m_{s.d}=m_s$ mit $d=D_s=1$ sind Kopien der Basismarke oder unstrukturierten Marke für die Markenart s=0.
- Die Kopien $m_{s.d}$ sind Kopien einer Attributmarke für die Markenarten $s\in\{1,...,A\}$ mit $A\in\mathcal{N}_+$.
- Die Kopien $m_{s.d}$ sind Kopien einer atomaren oder einfachen Marke für die Markenarten $s\in\{0,1,...,A\}$ mit $A\in\mathcal{N}_0$.
- Die Kopien $m_{s.d}$ sind Kopien einer Kompositmarke oder komplexen Marke für die Markenarten $s\in\{A+1,...,A+B\}$ mit $B\in\mathcal{N}_+$.
- Die Kopien $m_{s.d}$ sind Kopien einer strukturierten Marke für die Markenarten $s\in\{1,...,A+B\}$ mit $(A+B)\in\mathcal{N}_+$.
- Die Kopien $m_{s.d}$ sind Kopien einer sortierten Marke für die Markenarten $s\in\{0,...,A+B\}$ mit $(A+B)\in\mathcal{N}_0$.

N) Das algebraische Universum OB einer Spezifikation $SPEC_{MSIG}$ ist in konventioneller Weise als Vereinigungsmenge aller Mengen OB_i formaler Objekte aus der Familie OBF definiert:

$$OB = \bigcup (i\in\{1,...,I\}): OB_i$$

O) Die Markenontologie einer Spezifikation $SPEC_{MSIG}$ ist der Komplex, der aus ihrem algebraischen Universum OB und ihrer sortierten Markenmenge SMM resultiert. In der sortierten Markenmenge sind alle Informationen über die Sortenstruktur enthalten, die dem algebraischen Universum OB durch Attribut- und Markensorten überlagert wird. Diese Sorten legen fest, ob und - im positiven Fall - auf welche Weise die formalen Objekte aus dem algebraischen Universum intern strukturiert sind. Da die Attributsorten dabei nur den subsidiären Charakter besitzen, Markensorten aufzubauen[17)], dominiert insgesamt das Markenkonzept bei der Sortierung des algebraischen Universums. Daher wird auch von einer markenbezogenen Sortenstruktur gesprochen, die jede Spezifikation $SPEC_{MSIG}$ auf der Grundlage einer Markensignatur MSIG auszeichnet.

P) Jede Kopie m_0 der Basismarke ist als Konstante ein Grundterm aus der teilevaluierten Grundtermmenge $GTM_i=TTM_i(\emptyset)$ für die Sorte $sort_i$=bas_marke mit $i=Q+1$. Jede Kopie m_s einer strukturierte Marke stellt ein Grundtermtupel dar[18)]. Die Komponenten des Grundtermtupels sind jeweils Elemente aus einer teilevaluierten Grundtermmenge $GTM_i=TTM_i(\emptyset)$ für eine Sorte $sort_i$=str_marke$_s$ mit $i=Q+A+s+1$.

Q) Oftmals spielt es in Argumentationen keine wesentliche Rolle, zu differenzieren zwischen einerseits Marken, die im Kontext von Sorten und Operationssymbolen definiert sind, und andererseits Markenkopien, die sich auf formale Objekte und Operationen beziehen. Dann werden Marken und deren Kopien undifferenziert als "Marken" angesprochen.

R) Die Prädikatssymbolemenge PRÄ umfaßt unmittelbar nur die Namen $Prä_u$ aller spezifizierten Prädikatssymbole $Prä_u(sor\_marke_{s(u.1)}\ ...\ sor\_marke_{s(u.Ku)})$. Es wurde jedoch früher festgelegt, stets eindeutig identifizierende Prädikatssymbolnamen zu verwenden. Daher umfaßt die Prädikatssymbolemenge PRÄ in Verbindung mit den Spezifikationen der Prädikatssymbole in der Sektion "Präs" mittelbar alle Prädikatssymbole, die in einer markenbezogenen algebraisch-prädikatenlogischen Spezifikation definiert werden. Hierfür wird mit $u\in\{1,...,U\}$ und $U\in\mathcal{N}_+$ vorausgesetzt, daß stets mindestens ein Prädikatssymbol $Prä_u$ definiert ist.

S) Die Argumente von K_u-stelligen Prädikatssymbolen $Prä_u$ mit $K_u \in \mathcal{N}_+$ bestehen immer aus Markensorten. Den Regelfall bei den später abgeleiteten Netzmodellen werden einstellige Prädikatssymbole $Prä_u(sor\_marke_{s(u)})$[19] mit $K_u=1$ bilden.

T) Die Argumente von K_u-stelligen atomaren prädikatenlogischen Formeln $prä_u$, die jeweils einem Prädikatssymbol $Prä_u$ zugeordnet sind, sind stets aus Kopien sortierter Marken aufgebaut.

U) Durch ihre direkte Bezugnahme auf Markenkopien unterscheiden sich die prädikatenlogischen Formeln[20] aus einer markenbezogenen Spezifikation $SPEC_{MSIG}$ von den entsprechenden Prädikatsdefinitionen aus Prädikat/Transition-Netzen und verwandten Höheren Netzen. Denn diese alternativen Netzklassen sprechen in den Argumenten ihrer Prädikate nicht explizit Markenkopien an, sondern prädikatenlogische "Individuen". Diese "Individuen" besitzen aber - wie schon früher erläutert wurde - nicht die Qualität von Marken, sondern die von Attributausprägungen.

V) Prädikatssymbole und daraus formierte prädikatenlogische Formeln können sich in markenbezogenen Spezifikationen $SPEC_{MSIG}$ niemals unmittelbar auf Attribute bzw. Attributausprägungen beziehen. Ihre Argumente können unmittelbar nur aus Markensorten bzw. Markenkopien zusammengesetzt sein. Ein mittelbarer Bezug auf Attribute bzw. Attributausprägungen erfolgt jedoch im allgemeinen[21] dadurch, daß alle strukturierten Marken letztlich aus Attributmarken bestehen, die ihrerseits aus Attributen konstituiert sind.

W) Prädikatssymbole und prädikatenlogische Formeln besitzen in ihren Argumenten zumeist[22] eine komplexe innere Struktur. Sie besteht aus mindestens zwei Hierarchiestufen. Auf der ersten Stufe bestehen die Argumente aus Markensorten bzw. Markenkopien. Auf der letzten - untersten - Hierarchiestufe werden die Argumente durch Operationssymbole und Attributsorten bzw. durch Operationen und Attributausprägungen konstituiert. Darüber hinaus lassen sich beliebige weitere, dazwischenliegende Hierarchiestufen bilden, indem die Markensorten bzw. Markenkopien auf der jeweils oberen Hierarchiestufe aus anderen Markensorten bzw. Markenkopien zusammengesetzt werden. Auf diese Weise lassen sich für Prädikatssymbole und daraus formierte prädikatenlogische Formeln beliebig tief verschachtelte Argumentstrukturen aus Markensorten, Operationssymbolen und Attributsorten bzw. aus Markenkopien, Operationen und Attributausprägungen gestalten.

X) Attribut- und Markensorten stellen zueinander komplementäre Konstrukte dar. Gleiches gilt für die zugehörigen Attributausprägungen bzw. Markenkopien. Markensorten, Attributsorten und Operationssymbole gehören der Sphäre der zugrundeliegenden Signatur MSIG an. Markenkopien, Attributausprägungen und Operationen gehören dagegen zum Bereich der zugeordneten MSIG-Algebra.

Y) Die Übergangsoperationen tr_v, die das Übergangsschema ÜS der früher entfalteten operationalen Semantik in der MSIG-Spezifikation konkretisieren, werden von nun an als Transaktionen angesprochen. Ihre Bezeichnung erinnert an die spätere Implementierung der Übergangsoperationen tr_v mit Hilfe der Programmiersprache PROLOG: Dort nehmen sie die Gestalt von Transaktionen aus dem Konzept der (trans)aktionsorientierten Datenverarbeitung an.

Z) Jede Transaktion tr_v wird mit einem 7-Tupel identifiziert. Dieses 7-Tupel legt den Vor-, den Informations- und den Nachbereich der Transaktion durch die Auflistung der jeweils zugehörigen Prädikatssymbole fest. Für alle diese Prädikatssymbole werden - nach den drei vorgenannten Bereichen getrennt - Multimengen aus teilevaluierten atomaren Formelvorkommnissen aufgeführt. Schließlich werden einer Transaktion jene Restriktionsformeln explizit zugeordnet, die nicht schon in der Standardformelmenge $FOR_{standard}$ implizit enthalten sind. Das 7-Tupel, das diese transaktionsspezifischen Informationen ausdrückt, spezifiziert die innere Struktur der

Transaktion tr_v. Das Symbol "tr_v", das die derart spezifizierte Transaktion bezeichnet, wird auch als Transaktionsname angesprochen. Der Transaktionsname kann in konkreten Netzen anstelle des formalsprachlichen Symbols "tr_v" auch ein beliebiger[23)] natürlichsprachlicher Ausdruck sein.

α) Transaktionen bewirken[24)] durch ihre Entfernungs- und Ergänzungsanweisungen nur das Eliminieren bzw. Hinzufügen von ganzen Markenkopien. Sie führen niemals zur Veränderung der Attribut- oder Zusammensetzungsstruktur einer betroffenen Marke[25)]. Durch diese Invarianz der Markenstruktur wird die Erfüllung des Strukturerhaltungspostulats sichergestellt. Zugleich wird auf diese Weise die oben kritisierte Freizügigkeit des Signaturkonzepts entscheidend eingeschränkt. Dadurch wird die strukturelle Integrität der später abgeleiteten Synthetischen Netze und Netzmodelle per constructionem gewährleistet.

β) Die Transaktionenmenge TR enthält die Namen aller Transaktionen tr_v, die in einer markenbezogenen algebraisch-prädikatenlogischen Spezifikation definiert sind. Mit $v \in \{1,...,V\}$ und $V \in \mathcal{N}_+$ wird vorausgesetzt, daß immer mindestens eine Transaktion definiert ist. Die internen Strukturen der Transaktionen werden in der Sektion "trans" in der oben erläuterten Weise spezifiziert. Daher umfaßt die Transaktionenmenge TR nicht nur unmittelbar alle Transaktionsnamen, sondern in Verbindung mit der Sektion "trans" auch mittelbar alle Transaktionsspezifikationen.

χ) Das allgemeine Übergangsschema ÜS stimmt genau mit dem Schema für Übergangsoperationen überein, das bereits an früherer Stelle ausführlich erläutert wurde. Es wird hier nicht identisch wiederholt, da dies nur die Spezifikationsdarstellung aufblähte, ohne irgendeinen Erkenntnisgewinn zu bewirken.

δ) Das Definitionsschema für MSIG-Spezifikationen erscheint auf den ersten Blick als recht aufwendiger Formalismus. Dies gilt allerdings nur so lange, wie die schematische Spezifikationsdefinition allgemeingültig formuliert wird. Bei der späteren Spezifizierung konkreter Netzmodelle läßt sich die Definition der jeweils zugrundeliegenden MSIG-Spezifikationen deutlich vereinfachen. Beispielsweise braucht in der Deklaration eines Prädikatssymbols nicht auf die Markensorte "str_marke" Bezug genommen werden, wenn von vornherein bekannt ist, daß die entsprechende Stelle des Prädikatssymbols von einer Subsorte - etwa der Markensorte "att_marke" - eingenommen wird. Falls in einem solchen Fall in der Prädikatsdeklaration direkt die Subsorte "att_marke" notiert wird, entfällt die Vermittlung zwischen dieser Subsorte und der früheren Sorte "str_marke" durch das Symbol "Sor_s" der identischen Abbildung. Zugleich steigt der Informationswert der Deklaration des Prädikatssymbols.

<u>Beispiele für sortierte Marken und deren Kopien:</u>

- m_0, $marke_0()$ und $\varnothing$ sind unterschiedlich notierte, aber identisch behandelte Kopien der Basismarke.
- Eine Attributmarke att_marke_2, die als die Zielsorte eines Operationssymbols $Marke_2$ mit $Marke_2$: $attribut_5\ attribut_7\ attribut_{13} \rightarrow att\_marke_2$ definiert ist, besteht aus den Attributen der Argumentsorten $attribut_5$, $attribut_7$ und $attribut_{13}$.
- $m_2 = marke_2(at_{2.1}, at_{2.2}, at_{2.3})$ ist eine Kopie der Attributmarke att_marke_2. Die Markenkopie ist aus den Attributausprägungen $at_{2.1}$, $at_{2.2}$ und $at_{2.2}$ zusammengesetzt. Sie erfüllen die Beziehungen $at_{2.1} \in OB_{att.5}$, $at_{2.2} \in OB_{att.7}$ bzw. $at_{2.3} \in OB_{att.13}$.
- Die Attributmarke att_marke_2 wird durch ein weiteres Operationssymbol $Struk_2$ mit $Struk_2$: $att\_marke_2 \rightarrow str\_marke_2$ als eine strukturierte Marke der Sorte str_marke_2 ausgewiesen. Dem Operationssymbol $Struk_2$ ist die identische Abbildung $struk_2$ mit $struk_2$: $m_2 \rightarrow struk_2(m_2) = m_2$ zugeordnet. Jede Kopie der strukturierten Marke str_marke_2 fällt daher mit einer identischen Kopie der Attributmarke att_marke_2 zusammen.
- att_marke_5 ist eine zweite Attributmarke. Sie ist als die Zielsorte eines Operationssymbols $Marke_5$ mit $Marke_5$: $attribut_1\ attribut_6 \rightarrow att\_marke_5$ definiert. Sie besteht daher aus den Attributen der Argumentsorten $attribut_1$ und $attribut_6$.
- $m_5 = marke_5(at_{5.1}, at_{5.2})$ ist eine Kopie der Attributmarke att_marke_5. Die Markenkopie ist aus den Attributausprägungen $at_{5.1}$ und $at_{5.2}$ zusammengesetzt. Dafür gelten die beiden Beziehungen $at_{5.1} \in OB_{att.1}$ bzw. $at_{5.2} \in OB_{att.6}$.
- Die Attributmarke att_marke_5 wird durch ein weiteres Operationssymbol $Struk_5$ mit $Struk_5$: $att\_marke_5 \rightarrow str\_marke_5$ als eine strukturierte Marke der Sorte str_marke_5 ausgewiesen. Dem Operationssymbol $Struk_5$ ist die identische Abbildung $struk_5$ mit $struk_5$: $m_5 \rightarrow struk_5(m_5) = m_5$ zugeordnet. Jede Kopie der strukturierten Marke str_marke_5 fällt daher mit einer identischen Kopie der Attributmarke att_marke_5 zusammen.
- str_marke_6 ist eine 2-stellige Kompositmarke. Sie ist als die Zielsorte des Operationssymbols $Struk_6$ mit $Struk_6$: $str\_marke_2\ str\_marke_5 \rightarrow str\_marke_6$ definiert. Die Kompositmarke str_marke_6 ist daher aus der strukturierten Marke str_marke_2 und der strukturierten Marke str_marke_5 zusammengesetzt. Dem Operationssymbol $Struk_6$ ist die Operation $struk_6$ mit $struk_6$: $(m_2, m_5) \rightarrow struk_6(m_2, m_5) = m_6$ zugeordnet.
- $m_6 = struk_6(struk_2(marke_2(at_{2.1}, at_{2.2}, at_{2.3})), struk_5((marke_5(at_{5.1}, at_{5.2})))$ ist eine Kopie der Kompositmarke str_marke_6. Diese Markenkopie ist aus der Kopie $m_2 = struk_2(marke_2(at_{2.1}, at_{2.2}, at_{2.3}))$ der strukturierten Marke str_marke_2 und der Kopie $m_5 = struk_5(marke_5(at_{5.1}, at_{5.2}))$ der strukturierten Marke str_marke_5 zusammengesetzt.
- Bei den Strukturierungsoperationen $struk_2$ und $struk_5$ handelt es sich jeweils um die identische Abbildung. Daher kann die Kopie m_6 der Kompositmarke str_marke_6 vereinfacht dargestellt werden als: $struk_6(marke_2(at_{2.1}, at_{2.2}, at_{2.3}), marke_5(at_{5.1}, at_{5.2}))$.
- Schließlich kann auf die o.a. Definitionsgleichungen $m_2 = marke_2(at_{2.1}, at_{2.2}, at_{2.3})$ und $m_5 = marke_5(at_{5.1}, at_{5.2})$ für die beiden Markenkopien m_2 und m_5 zurückgegriffen werden. Mit ihrer Hilfe läßt sich die Darstellung der Kopie m_6 der Kompositmarke noch weiter vereinfachen zu: $m_6 = struk_6(m_2, m_5)$.

Aufgrund ihrer Kompaktheit und Übersichtlichkeit werden in dieser Arbeit die beiden zuletzt angeführten vereinfachten Notationen für Markenkopien bevorzugt[26).

Anmerkungen zum Kapitel:

1) Sie werden auch kurz als markenbezogene Spezifikationen bezeichnet.

2) Beispielsweise besitzt eine Attributmarke, die durch die Sorte "att_marke" repräsentiert wird, den Namen "att_marke".

3) Ebenso läßt sich die Attributsorte als Name des repräsentierten Attributs verwenden.

4) Das konstituierte Attribut wird dann auch als einfaches oder atomares Attribut bezeichnet.

5) Aus prädikatenlogischem Blickwinkel stellt jedes dieser atomaren formalen Objekte eine Konstante dar.

6) Da in dieser Arbeit nur zulässige Attributausprägungen betrachtet werden, können sie auch kurz als Attributausprägungen angesprochen werden.

7) Das konstituierte Attribut heißt dann auch ein komplexes oder zusammengesetztes Attribut.

8) Da in dieser Arbeit nur zulässige Attributausprägungen betrachtet werden, können sie auch kurz als Attributausprägungen angesprochen werden.

9) Dann können die Attributausprägungen je nach der aktuell betrachteten Attributdefinition entweder als atomare oder aber als zusammengesetzte formale Objekte behandelt werden.

10) Das ist der schon früher erörterte Fall einer multiplen Zielsortendefinition.

11) Für A=0 existieren keine Attributmarken und a fortiori auch nicht deren Kopien. Dann sind mit $J_A=0$ auch keine Operationssymbole "Marke_j" definiert.

12) Wenn dies der Fall ist, liegt eine Realisierung der schon früher vorgestellten multiplen Zielsortendefinition vor. Dann gilt $J_A>A$.

13) Daher ist es möglich, für Kompositmarken - analog zu Attributmarken - ebenfalls multiple Markendefinitionen vorzunehmen. Für B=0 gilt wiederum $J_B=0$.

14) Die Argumente ihrer Operationssymbole "Struk_j" mit $j\in\{A+1,...,A+B\}$ können mit Hilfe der Operationssymbole "Marke_j" mit $j\in\{1,...,A\}$ so transformiert werden, daß sie sich ausschließlich auf Markensorten "$\text{att\_marke}_{s(j)}$" für Attributmarken erstrecken.

15) Die Argumente ihrer Operationssymbole "Struk_j" mit $j\in\{A+1,...,A+B\}$ können mit Hilfe der Operationssymbole "Marke_j" mit $j\in\{1,...,A\}$ niemals so transformiert werden, daß sie sich nur noch auf Markensorten "$\text{att\_marke}_{s(j)}$" für Attributmarken erstrecken würden.

16) Dies gilt uneingeschränkt für alle strukturierten Marken mit s>0. Nur die Kopien m_s der Basismarke mit s=0 müssen immer identisch sein.

17) Dieser subsidiäre Attributcharakter äußert sich z.B. darin, daß nur Marken formale Objekte darstellen, die - im Rahmen der hier entfalteten Markenontologie - selbständig existieren können. Attribute werden dagegen als formale Objekte ohne eigenständige Existenzmöglichkeit konzeptualisiert. Daher können Attribute niemals "an sich", sondern nur "für eine Marke" definiert werden.

18) Vgl. REISIG (1989a), S. 40 u. 44. Allerdings spricht REISIG von Grundtermen und nicht von Grundtermtupeln. Dies liegt daran, daß REISIG - wie sonst üblich - die Argumente von Prädikaten nicht aus Markensorten und -kopien, sondern aus prädikatenlogischen Individuen aufbaut.

19) Die Notationen $\text{Prä}_u(\text{sor\_marke}_{s(u.1)})$ und $\text{Prä}_u(\text{sor\_marke}_{s(u.Ku)})$ können für $K_u=1$ zu $\text{Prä}_u(\text{sor\_marke}_{s(u)})$ zusammengezogen werden.

20) Prädikatssymbole werden in Prädikat/Transition-Netzen und verwandten Höheren Netzen im allgemeinen nicht berücksichtigt, zumindest nicht klar von den daraus formierten prädikatenlogischen Formeln unterschieden.

21) Ein Bezug auf Attribute und deren Ausprägungen fehlt nur in dem Sonderfall, in dem sich die Argumente eines Prädikatssymbols und der zugehörigen prädikatenlogischen Formeln nur auf die Sorte "bas_marke" bzw. auf Kopien der Basismarke m_0 erstrecken. Dies bedeutet in Modellierungskontexten, daß keine Eigenschaften der jeweils modellierten realen Objekte interessieren, sondern nur die Existenz dieser Objekte bedeutsam ist. Nachfolgend wird auf diesen Sonderfall nicht weiter eingegangen.

22) Die nachfolgenden Ausführungen setzen voraus, daß der Sonderfall, der in der voranstehenden Anmk. angesprochen wurde, nicht vorliegt. Er werden also nur solche Prädikatssymbole und zugehörige prädikatenlogische Formeln betrachtet, deren Argumente - direkt oder indirekt - mindestens einmal eine Sorte "att_marke_a" bzw. die Kopie m_a einer Attributmarke mit $a\in\{1,...,A\}$ umfassen.

23) Es gilt lediglich zwei einschränkende Bedingungen zu beachten: Erstens müssen diese Ausdrücke einen eindeutig identifizierenden Charakter besitzen. Die Transaktionsnamen dürfen also mit keiner anderen Netzkomponente verwechselt werden können. Dies ist für die spätere Implementierung von Netzmodellen mit Hilfe der Automatischen Informationsverarbeitung erforderlich. Zweitens müssen die Ausdrücke jeweils mit einem Kleinbuchstaben beginnen. Dies folgt aus der Modellimplementierung durch die Programmiersprache PROLOG. Sie fordert, daß nur jene Ausdrücke, die Variablen darstellen, mit einem Großbuchstaben beginnen dürfen. Bei Transaktionsnamen handelt es sich aber nicht um Variablen, sondern um Konstanten.

24) Dabei handelt es sich nur um mittelbare Wirkungen, weil die direkte Wirkung in dem Eliminieren oder Hinzufügen von faktischen Prädikatsvorkommnissen besteht. Diese Prädikatsvorkommnisse enthalten aber in ihren Argumenten diejenigen Markenkopien, deren argumentspezifische Kombination das zugehörige Prädikat jeweils erfüllt. Daher bedeutet das Eliminieren bzw. Hinzufügen faktischer Prädikatsvorkommnisse mittelbar auch das entsprechende Eliminieren bzw. Hinzufügen der jeweils involvierten Markenkopien.

25) Diese Anforderung folgt mittelbar bereits aus der voranstehenden Anforderung an MSIG-Prädikate und aus der Definition des Übergangsschemas ÜS: Da die Entfernungs- sowie Ergänzungsanweisungen aus dem Eliminieren bzw. Hinzufügen faktischer Prädikatsvorkommnisse bestehen und weil sich die Argumente dieser Prädikatsvorkommnisse unmittelbar nur auf Markenkopien erstrecken dürfen, bewirken die Anweisungsausführungen auch das Eliminieren bzw. Hinzufügen der Markenkopien aus den Argumenten von faktischen Prädikatsvorkommnissen.

26) Dies ist allerdings nur dort möglich, wo Informationen über die konkreten Markenstrukturen für die Reduzierung der Darstellungskomplexität genutzt werden können.

5.1.2 Präzisierung des Kernkonzepts Synthetischer Netze

5.1.2.1 Definition Synthetischer Netze

Das Kernkonzept Synthetischer Netze umfaßt dasjenige Teilkonzept, das sich allein durch die Einführung sortierter Marken erklären läßt. Es beruht auf Signaturkonzept und Prädikatenlogik, die miteinander kombiniert, erweitert und schließlich zu den markenbezogenen algebraisch-prädikatenlogischen MSIG-Spezifikationen verdichtet wurden. Die zuvor definierten und erläuterten formalen Konstrukte werden nun in einem Definitionsschema[1)] für Synthetische Netze zusammengeführt[2)].

Definition: Synthetisches Netz

Ein Synthetisches Netz ist - unter dem Vorbehalt späterer Modifizierungen - ein geordnetes, zweistufig-hierarchisches 5-Tupel $SN=(TOP,SPEC_{MSIG};BES,M_0;IB)$, für das gilt:

1) $TOP=(S,T;F)$ ist die Netztopologie. Sie stellt ein konventionell definiertes Petrinetz i.e.S. dar:

- ❑ Die Stellenmenge $S=\{s_m: m=1,...,M\}$ mit $M\in \mathcal{N}_+$ ist eine endliche, nicht-leere Menge aus atomaren formalen Objekten s_m der Objektart "Stelle".
- ❑ Die Transitionenmenge $T=\{t_n: n=1,...,N\}$ mit $N\in \mathcal{N}_+$ ist eine endliche, nicht-leere Menge aus atomaren formalen Objekten t_n der Objektart "Transition".
- ❑ Die Flußrelation $F\subseteq((SxT)\cup(TxS))$ ist eine endliche, nicht-leere Menge von zusammengesetzten formalen Objekten, die jeweils geordnete Paare (kn_x,kn_y) aus artverschiedenen atomaren formalen Objekten darstellen.
- ❑ Die Netztopologie beruht auf der Auszeichnung lokaler Umgebungen von Stellen und Transitionen als deren Vor- und Nachbereiche VB bzw. NB. Hierfür gilt:

$$\forall(s_m\in S):\ VB(s_m) = \{t_n: t_n\in T \wedge (t_n,s_m)\in F\}$$
$$\wedge\quad NB(s_m) = \{t_n: t_n\in T \wedge (s_m,t_n)\in F\}$$
$$\forall(t_n\in T):\ VB(t_n) = \{s_m: s_m\in S \wedge (s_m,t_n)\in F\}$$
$$\wedge\quad NB(t_n) = \{s_m: s_m\in S \wedge (t_n,s_m)\in F\}$$
$$VB(F) = \{kn_x\in(S\cup T): \exists(kn_y\in(S\cup T)): (kn_x,kn_y)\in F\}$$
$$NB(F) = \{kn_y\in(S\cup T): \exists(kn_x\in(S\cup T)): (kn_x,kn_y)\in F\}$$

Die graphische Repräsentation der Netztopologie ist ein bipartiter gerichteter Graph $GR=(KN,KA)$ mit Knoten "kn" aus der Knotenmenge $KN=(S\ T)$ und gerichteten Kanten (kn_x,kn_y) aus der Kantenmenge $KA=F$. Knoten aus der Stellenmenge S (Transitionenmenge T) heißen stellenartige (transitionenartige) Knoten.

2) $SPEC_{MSIG}=(SO,OP,PRÄ;OBF,OPF,FAK_0;TTMF(VAF),VBM;RES;TR;ÜS)$ ist die Netzspezifikation. Es handelt sich um eine markenbezogene algebraisch-prädikatenlogische Spezifikation bezüglich der Markensignatur $MSIG=(SO,OP,PRÄ)$. Die MSIG-Spezifikation erfüllt folgende schematische Deklaration:

Netzspezifikation =

sorts: $sort_1 = attribut_1$

...

$sort_Q = attribut_Q$

$sort_{Q+1} = bas\_marke$

$sort_{Q+2} = att\_marke_1$

...

$sort_{Q+A+1} = att\_marke_A$

$sort_{Q+A+2} = str\_marke_1$

...

$sort_{Q+2A+B+1} = str\_marke_{A+B}$

$sort_{Q+2A+B+2} = sor\_marke_0$

...

$sort_{Q+3A+2B+2} = sor\_marke_{A+B}$

$SO = \{sort_i: i=1,...,I\}$ mit $I = Q+3A+2B+2$

Ops: $Op_1 = Att_1$: $\rightarrow attribut_{q(1)}$

...

$Op_{JQ} = Att_{JQ}$: $attribut_{q(JQ.1)} \dots attribut_{q(JQ.KJQ)} \rightarrow attribut_{q(JQ.KJQ+1)}$

$Op_{JQ+1} = Marke_0$: $\rightarrow bas\_marke$

$Op_{JQ+2} = Marke_1$: $attribut_{q(JQ+2.1)} \dots attribut_{q(JQ+2.KJQ+2)} \rightarrow att\_marke_{s(1)}$

...

$Op_{JQ+JA+1} = Marke_{JA}$: $attribut_{q(JQ+JA+1.1)} \dots attribut_{q(JQ+JA+1.KJQ+JA+1)} \rightarrow att\_marke_{s(JA)}$

$Op_{JQ+JA+2} = Struk_1$: $att\_marke_1 \rightarrow str\_marke_1$

...

$Op_{JQ+A+JA+1} = Struk_A$: $att\_marke_A \rightarrow str\_marke_A$

$Op_{JQ+A+JA+2} = Struk_{A+1}$: $str\_marke_{s(A+1.1)} \dots str\_marke_{s(A+1.KA+1)} \rightarrow str\_marke_{s(A+1.KA+1+1)}$

...

$Op_{JQ+A+JA+JB+1} = Struk_{A+JB}$: $str\_marke_{s(A+JB.1)} \dots str\_marke_{s(A+JB.KA+JB)} \rightarrow str\_marke_{s(A+JB.KA+JB+1)}$

$Op_{JQ+A+JA+JB+2} = Sor_0$: $bas\_marke \rightarrow sor\_marke_0$

$Op_{JQ+A+JA+JB+3} = Sor_1$: $str\_marke_1 \rightarrow sor\_marke_1$

...

$Op_{JQ+2A+B+JA+JB+2} = Sor_{A+B}$: $str\_marke_{A+B} \rightarrow sor\_marke_{A+B}$

$OP = \{Op_j : j = 1,...,J\}$ mit $J = J_Q + J_A + J_B + 2 + 2A + B$

<u>OBs</u>:

$OB_1 = OB_{att.1}$
...
$OB_Q = OB_{att.Q}$

$OB_{Q+1} = OB_{bas\_marke} = SYMBOL$
$OB_{Q+1+a} = OB_{att\_marke} = SYMBOL$ für $a \in \{1,...,A\}$
$OB_{Q+1+A+b} = OB_{str\_marke} = SYMBOL$ für $b \in \{1,...,A+B\}$
$OB_{Q+1+2A+B+s} = OB_{sor\_marke} = SYMBOL$ für $s \in \{0,...,A+B\}$

$OBF = (OB_i : i = 1,...,I)$

<u>ops</u>:

$op_1 = att_1$: $\rightarrow OB_{att.q(1)}$
$att_1() = at_1$
...
$op_{JQ} = att_{JQ}$: $OB_{att.q(JQ.1)} \times ... \times OB_{att.q(JQ.KJQ)} \rightarrow OB_{att.q(JQ.KJQ+1)}$
$(at_{JQ.1},...,at_{JQ.KJQ}) \rightarrow att_{JQ}(at_{JQ.1},...,at_{JQ.KJQ}) = at_{JQ.KJQ+1}$

$op_{JQ+1} = marke_0$: $\rightarrow OB_{bas\_marke}$
$marke_0() = \varnothing = m_0$

$op_{JQ+2} = marke_1$: $OB_{att.q(JQ+2.1)} \times ... \times OB_{att.q(JQ+2.KJQ+2)} \rightarrow OB_{att\_marke}$
$(at_{JQ+2.1},...,at_{JQ+2.KJQ+2})$
$\rightarrow marke_1(at_{JQ+2.1},...,at_{JQ+2.KJQ+2}) = m_{s(1)}$
...
$op_{JQ+JA+1}$
$= marke_{JA}$: $OB_{att.q(JQ+JA+1.1)} \times ... \times OB_{att.q(JQ+JA+1.KJQ+JA+1)} \rightarrow OB_{att\_marke}$
$(at_{JQ+JA+1.1},...,at_{JQ+JA+1.KJQ+JA+1})$
$\rightarrow marke_A(at_{JQ+JA+1.1},...,at_{JQ+JA+1.KJQ+JA+1}) = m_{s(JA)}$

$op_{JQ+JA+2}$
$= struk_1$: $OB_{att\text{-}marke} \rightarrow OB_{str\_marke}$
$m_1 \rightarrow struk_1(m_1) = m_1$
...
$op_{JQ+A+JA+1}$
$= struk_A$: $OB_{att\text{-}marke} \rightarrow OB_{str\_marke}$
$m_A \rightarrow struk_A(m_A) = m_A$

$op_{JQ+A+JA+2}$
$= struk_{A+1}$: $OB_{str\_marke} \times ... \times OB_{str\_marke} \rightarrow OB_{str\_marke}$
$(m_{s(A+1.1)},...,m_{s(A+1.KA+1)}) \rightarrow ...$
$struk_{A+1}(m_{s(A+1.1)},...,m_{s(A+1.KA+1)}) = m_{s(A+1.KA+1+1)}$
...
$op_{JQ+A+JA+JB+1}$
$= struk_{A+JB}$: $OB_{str\_marke} \times ... \times OB_{str\_marke} \rightarrow OB_{str\_marke}$
$(m_{s(A+JB.1)},...,m_{s(A+JB.KA+JB)}) \rightarrow ...$
$struk_{A+JB}(m_{s(A+JB.1)},...,m_{s(A+JB.KA+JB)}) = m_{s(A+JB.Ka+JB+1)}$

$op_{JQ+A+JA+JB+2}$
$= sor_0$: $OB_{bas\_marke} \rightarrow OB_{sor\_marke}$
$m_0 \rightarrow sor_0(m_0) = m_0$

$op_{JQ+A+JA+JB+3}$ $\quad OB_{str\_marke} \rightarrow OB_{sor\_marke}$
$= sor_1$: $\quad m_1 \rightarrow sor_1(m_1) = m_1$
...
$op_{JQ+2A+B+JA+JB+2}$ $\quad OB_{str\_marke} \rightarrow OB_{sor\_marke}$
$= sor_{A+B}$: $\quad m_{A+B} \rightarrow sor_{A+B}(m_{A+B}) = m_{A+B}$

$OPF = (op_j: j=1,...,J)$ mit $J = J_Q+J_A+J_B+2+2A+B$

<u>präs</u>: präi$_1$:

präs: $prä_1$: $\quad OB_{sor\text{-}marke} \times ... \times OB_{sor\_marke}$
$fakt_0(mu_{1.0.1}, prä_1(m_{s(1.1.1)},...,m_{s(1.K1.1)}))$
...
$fakt_0(mu_{1.0.D1}, prä_1(m_{s(1.1.D1)},...,m_{s(1.K1.D1)}))$
$FAK_{1.0} = \{fakt_0(mu_{1.0.d}, prä_1(m_{s(1.1.d)},...,m_{s(1.K1.d)})): d=1,...,D_1\}$
...
$prä_U$: $\quad OB_{sor\text{-}marke} \times ... \times OB_{sor\_marke}$
$fakt_0(mu_{U.0.1}, prä_U(m_{s(U.1.1)},...,m_{s(U.KU.1)}))$
...
$fakt_0(mu_{U.0.DU}, prä_U(m_{s(U.1.DU)},...,m_{s(U.KU.DU)}))$
$FAK_{U.0} = \{fakt_0(mu_{U.0.d}, prä_U(m_{s(U.1.d)},...,m_{s(U.KU.d)})): d=1,...,D_U\}$

$FAK_0 = (FAK_{u.0}: u=1,...,U)$

<u>Vars</u>: $VA_1 = \{X_{1.1},...,X_{1.V1}\}$
...
$VA_I = \{X_{I.1},...,X_{I.VI}\}$

$VAF = (VA_i: i=1,...,I)$

<u>Vars</u>: $VA_1 = \{X_{1.1},...,X_{1.V1}\}$
...
$VA_I = \{X_{I.1},...,X_{I.VI}\}$

$VAF = (VA_i: i=1,...,I)$

<u>terms</u>: $KB_1 = \{te: \exists(op_j \in OPF): op_j: \rightarrow OB_1 \wedge op_j() = te\}$

$TTM_1(VA_1)$ minimal mit:

$$
\begin{aligned}
& KB_1 = \emptyset \rightarrow OB_1 \subseteq TTM_1(VA_1) \\
\wedge\ & KB_1 \neq \emptyset \rightarrow KB_1 \subseteq TTM_1(VA_1) \\
\wedge\ & VA_1 \subseteq TTM_1(VA_1) \\
\wedge\ & (\forall(op_j \in OPF): ... \\
& \quad (op_j: OB_{i(j.1)} \times ... \times OB_{i(j.Kj)} \rightarrow OB_{i(j.Kj+1)} \\
& \quad \wedge\ (\forall(k \in \{1,...,K_j\}): te_k \in TTM_{i(j.k)}(VA_{i(j.k)})) \\
& \quad \wedge\ i(j.K_j+1) = 1 \wedge op_j(te_1,...,te_{Kj}) = te_{Kj+1}) \\
& \rightarrow\ te_{Kj+1} \in TTM_1(VA_1))
\end{aligned}
$$

...

$KB_I = \{te: \exists(op_j \in OPF): op_j: \rightarrow OB_I \wedge op_j() = te\}$

$TTM_I(VA_I)$ minimal mit:

$$
\begin{aligned}
& KB_I = \varnothing \rightarrow OB_I \subseteq TTM_I(VA_I) \\
\wedge\ & KB_I \neq \varnothing \rightarrow KB_I \subseteq TTM_I(VA_I) \\
\wedge\ & VA_I \subseteq TTM_I(VA_I) \\
\wedge\ & (\forall(op_j \in OPF): \ldots \\
& \qquad (op_j: OB_{i(j.1)} \text{ x} \ldots \text{x } OB_{i(j.Kj)} \rightarrow OB_{i(j.Kj+1)} \\
& \qquad \wedge\ (\forall(k \in \{1,\ldots,K_j\}): te_k \in TTM_{i(j.k)}(VA_{i(j.k)})) \\
& \qquad \wedge\ i(j.K_j+1) = I \wedge op_j(te_1,\ldots,te_{Kj}) = te_{Kj+1}) \\
& \quad \rightarrow\ te_{Kj+1} \in TTM_I(VA_I))
\end{aligned}
$$

$TTMF(VAF) = (TTM_i(VA_i): i=1,\ldots,I)$

$$
\begin{aligned}
VBM = \{ & vb_c: c=1,\ldots,C \wedge C \in \mathcal{N}_+ \wedge (\forall(c \in \{1,\ldots C\}: K_c \in \mathcal{N}_+ \ldots \\
& vb_c: TTM_{i(1)}(VA_{i(1)}) \text{ x} \ldots \text{x } TTM_{i(Kc)}(VA_{i(Kc)}) \rightarrow OB_{i(1)} \text{ x} \ldots \text{x } OB_{i(Kc)})\}
\end{aligned}
$$

<u>res</u>: a) entweder für Z=0: $RES = FOR_{standard}$

b) oder aber für $Z \in \mathcal{N}_+$:

$for_1(te_1,\ldots,te_{K1})$

...

$for_z(te_1,\ldots,te_{Kz})$

mit:

$\forall(z \in \{1,\ldots,Z\})\ \exists(K_z \in \mathcal{N}_+)\ \forall(k \in \{1,\ldots,K_z\}): te_k \in TTM_{i(k)}(VA_{i(k)})$

$RES = \{for_z(te_1,\ldots,te_{Kz}): z=1,\ldots,Z\} \cup FOR_{standard}$

<u>res</u>: a) entweder für Z=0: $RES = FOR_{standard}$

b) oder aber für $Z \in \mathcal{N}_+$:

$for_1(te_1,\ldots,te_{K1})$

...

$for_z(te_1,\ldots,te_{Kz})$

mit:

$\forall(z \in \{1,\ldots,Z\})\ \exists(K_z \in \mathcal{N}_+)\ \forall(k \in \{1,\ldots,K_z\}): te_k \in TTM_{i(k)}(VA_{i(k)})$

$RES = \{for_z(te_1,\ldots,te_{Kz}): z=1,\ldots,Z\} \cup FOR_{standard}$

<u>trans</u>: $tr_1 = \ldots$

$(\ VB(tr_1) = \{Prä_u(sor\_marke_{s(u.1)},\ldots,sor\_marke_{s(u.Ku)}): u \in IVB_1\},$

$MTAV_1 = \{MTAV_{u.1}: u \in IVB_1\},$

$IB(tr_1) = \{Prä_u(sor\_marke_{s(u.1)},\ldots,sor\_marke_{s(u.Ku)}): u \in IIB_1\},$

$MTAI_1 = \{MTAI_{u.1}: u \in IIB_1\},$

$NB(tr_1) = \{Prä_u(sor\_marke_{s(u.1)},\ldots,sor\_marke_{s(u.Ku)}): u \in INB_1\},$

$MTAN_1 = \{MTAN_{u.1}: u \in INB_1\},$

$RES_1 \in \{\emptyset, \{for_{z(1.h)}: \exists(H_1 \in \mathcal{N}_+)\ \forall(h \in \{1,\ldots,H_1\}): for_{z(1.h)} \in RES\}\})$

...

$tr_V = \ldots$

$(\ VB(tr_V) = \{Prä_u(sor\_marke_{s(u.1)},\ldots,sor\_marke_{s(u.Ku)}): u \in IVB_V\},$

$MTAV_V = \{MTAV_{u.V}: u \in IVB_V\},$

$IB(tr_V) = \{Prä_u(sor\_marke_{s(u.1)},\ldots,sor\_marke_{s(u.Ku)}): u \in IIB_V\},$

$MTAI_V = \{MTAI_{u.V}: u \in IIB_V\},$

$NB(tr_V) = \{Prä_u(sor\_marke_{s(u.1)},\ldots,sor\_marke_{s(u.Ku)}): u \in INB_V\},$

$MTAN_V = \{MTAN_{u.V}: u \in INB_V\},$

$RES_V \in \{\emptyset, \{for_{z(V.h)}: \exists(H_V \in \mathcal{N}_+)\ \forall(h \in \{1,\ldots,H_V\}): for_{z(V.h)} \in RES\}\})$

$TR = \{tr_v: v = 1,\ldots,V\}$

mit:

$\forall(tr_v \in TR): \ldots$

$vb_c(MTAV_v) = (vb_c(MTAV_{u.v}): u \in IVB_v)$

$vb_c(MTAI_v) = (vb_c(MTAI_{u.v}): u \in IIB_v)$

$vb_c(MTAN_v) = (vb_c(MTAN_{u.v}): u \in INB_v)$

$EB_v = VB_v \cup IB_v \ \wedge\ IEB_v = IVB_v \cup IIB_v$

$MTAE_{u.v} \in \{MTAV_{u.v}, MTAI_{u.v}\}$

$MTAE_v = (MTAE_{u.v}: u \in IEB_v)$

$vb_c(MTAE_v) = (vb_c(MTAE_{u.v}): u \in IEB_v)$

$WB_v = VB_v \cup NB_v \ \wedge\ IWB_v = IVB_v \cup INB_v$

$MTAW_{u.v} \in \{MTAV_{u.v}, MTAN_{u.v}\}$

$MTAW_v = (MTAW_{u.v}: u \in IWB_v)$

$vb_c(MTAW_v) = (vb_c(MTAW_{u.v}): u \in IWB_v)$

$OB_v = VB_v \cup IB_v \cup NB_v \ \wedge\ IOB_v = IVB_v \cup IIB_v \cup INB_v$

$MTAO_{u.v} \in \{MTAV_{u.v}, MTAI_{u.v}, MTAN_{u.v}\}$

$MTAO_v = (MTAO_{u.v}: u \in IOB_v)$

$vb_c(MTAO_v) = (vb_c(MTAO_{u.v}): u \in IOB_v)$

$MTA = (MTA_{u.v}: u = 1,\ldots,U \wedge v = 1,\ldots,V \wedge \ldots$

$\quad MTA_{u.v} \in \{MTAV_{u.v}, MTAI_{u.v}, MTAN_{u.v}\})$

<u>ÜS</u>:

IF prätest($FAK_{u.r}$) = gültig für alle $u \in IIB_v$

THEN IF $FAK_{u.r} \geq vb_c(MTAE_{u.v})$ für alle $u \in IEB_v$

AND

haupttest($vb_c(MTAE_{u.v})$) = gültig für alle $u \in IEB_v$

AND

$vb_c(X_k) = vb_c(\text{bestimme}_k(X_{k.1},\ldots,X_{k.Hk}))$

mit: $\forall(h \in \{1,\ldots,H_k\})\ \exists(u \in IEB_v)$: $X_{k.h} \in VA(MTAE_{u.v})$

für alle $X_k \in VA(MTAN_{u.v})$

THEN DO Übergangsprozedur

DO Faktenentfernung

$FAK_{u.-} := FAK_{u.r} - vb_c(MTAV_{u.v})$ für alle $u \in IVB_v$

$FAK_{u.-} := FAK_{u.r}$ für alle $u \in (IPRÄ\text{-}IVB_v)$

ENDDO Faktenentfernung

DO Faktenergänzung

$FAK_{u.+} := FAK_{u.-} + vb_c(MTAN_{u.v})$ für alle $u \in INB_v$

$FAK_{u.+} := FAK_{u.-}$ für alle $u \in (IPRÄ\text{-}INB_v)$

ENDDO Faktenergänzung

ENDDO Übergangsprozedur

IF posttest($FAK_{u.+}$) = gültig für alle $u \in IWB_v$

THEN DO Übergangsakt

$FAK_{u.f} := FAK_{u.+}$ für alle $u \in IPRÄ$

übergang($vb_c(MTAO_v)$) := gültig

ENDDO Übergangsakt

ELSE UNDO Übergangsprozedur

$FAK_{u.f} := FAK_{u.r}$ für alle $u \in IPRÄ$

übergang($vb_c(MTAO_v)$) := ungültig

EUNDO Übergangsprozedur

ENDIF

ENDIF

ENDIF

3) BES = (bsp,bsk,btt,bfm) ist die Netzbeschriftung. Ihre Beschriftungsfunktionen "bsp", "bsk", "btt" und "bfm" bilden die Knoten und Kanten aus der Netztopologie auf Konstrukte aus der Netzspezifikation ab:

- Die prädikatsbezogene Stellenbeschriftung "bsp" bildet jede Stelle s_m aus der Menge S auf den Namen $Prä_u$ eines Prädikatssymbols aus der Menge PRÄ in bijektiver Weise[3] ab:

$$\begin{aligned} \text{bsp:} \quad & S \rightarrow PRÄ \\ & s_m \rightarrow bsp(s_m) = Prä_u \end{aligned}$$

- Die kapazitätsbezogene Stellenbeschriftung "bsk" bildet jede Stelle s_m aus der Menge S auf ihre Kapazität KAP_m ab. Die Stellenkapazität ist die maximal zulässige Anzahl von Fakten, die unter einer Markierung M_r in der Faktenmenge $FAK_{r.u}$ des Prädikatssymbols $Prä_u$ für die Stelle s_m enthalten sein dürfen:

$$\begin{aligned} \text{bsk:} \quad & S \rightarrow \mathcal{N}_+ \cup \{\omega\} \\ & s_m \rightarrow bsk(s_m) = KAP_m \end{aligned}$$

- Die Transitionenbeschriftung "btt" bildet jede Transition t_n aus der Menge T auf den Namen tr_v einer Transaktion aus der Menge TR in bijektiver Weise[4] ab:

$$\begin{aligned} \text{btt:} \quad & T \rightarrow TR \\ & t_n \rightarrow btt(t_n) = tr_v \end{aligned}$$

- Die Kantenbeschriftung "bfm" ordnet jeder Kante (s_m,t_n) und jeder Kante (t_n,s_m) aus der Flußrelation F eine Multimenge $MTAV_{u.v}$ oder $MTAI_{u.v}$ bzw. $MTAN_{u.v}$ aus teilevaluierten atomaren Formeln $prä_u(te_1,...,te_{Ku})$ zu[5]. Dies ist genau dann der Fall, wenn die kantenzugehörige Stelle s_m auf das Prädikatssymbol $Prä_u$ und die kantenzugehörige Transition t_n auf die Transaktion tr_v abgebildet worden sind und wenn das Prädikatssymbol $Prä_u$ zum Vor- oder Informations- bzw. zum Nachbereich der Transaktion tr_v gehört. Mit MTA als Familie aller Multimengen aus teilevaluierten atomaren Formeln, die in der Sektion "trans" der Netzspezifikation eingeführt wurden, gilt:

$$\text{bfm:} \quad ((SxT) \cup (TxS)) \rightarrow MTA$$

$$(s_m,t_n) \rightarrow bfm(s_m,t_n) = \begin{cases} MTAV_{u.v}; & \text{für } u \in IVB_v \wedge (s_m,t_n) \in F \wedge \ldots \\ & bsp(s_m) = Prä_u \wedge btt(t_n) = tr_v \\ \{\}; & \text{für } (s_m,t_n) \notin F \\ MTAI_{u.v}; & \text{für } u \in IIB_v \wedge (s_m,t_n) \in F \wedge \ldots \\ & bsp(s_m) = Prä_u \wedge btt(t_n) = tr_v \end{cases}$$

$$(t_n,s_m) \rightarrow bfm(t_n,s_m) = \begin{cases} MTAN_{u.v}; & \text{für } u \in INB_v \wedge (t_n,s_m) \in F \wedge \ldots \\ & bsp(s_m) = Prä_u \wedge btt(t_n) = tr_v \\ \{\}; & \text{für } (t_n,s_m) \notin F \end{cases}$$

Die Formelmultimengen $MTAV_{u.v}$, $MTAI_{u.v}$ und $MTAN_{u.v}$ heißen Kantengewichte für ihre zugrundeliegenden Kanten.

4) Die Markierungsfunktion oder Ausgangsmarkierung M_0 bildet jede Stelle s_m aus der Menge S genau dann auf die Faktenmultimenge $FAK_{u.0}$ ab, wenn der Stelle s_m ein Prädikatssymbol $Prä_u$ aus der Prädikatssymbolemenge PRÄ zugeordnet worden ist. Mit FAK_0 als Familie aller Faktenmultimengen $FAK_{u.0}$, die in der Netzspezifikation für die Prädikatssymbole $Prä_u$ eingeführt wurden, gilt:

$$
\begin{aligned}
M_0: \quad & S \rightarrow FAK_0 \\
& s_m \rightarrow M_0(s_m) = FAK_{u.0}; \text{ sofern } bsp(s_m) = Prä_u
\end{aligned}
$$

5) $IB = \{IB_D, IB_E; IB_V, IB_I, IB_W; IB_B; IB_0, IB_G\}$ ist die Menge aller netzkonstitutiven Integritätsbedingungen.

- Die Disjunktheitsbedingung IB_D, Existenzbedingung IB_E und Verknüpftheitsbedingung IB_V zeichnen die Netztopologie TOP als ein Petrinetz aus:

IB_D: $S \cap T = \emptyset$

IB_E: $S \cup T \neq \emptyset$

IB_V: $S \cup T = VB(F) \cup NB(F)$

- Die Informationsbedingung IB_I trennt die Informationsbereiche der Transaktionen von ihren Vor- und Nachbereichen:

$\forall(tr_v \in TR)$: $VB_v \wedge IB_v = \emptyset \;\wedge\; NB_v \wedge IB_v = \emptyset$

- Die Wirkungsbedingung IB_W schließt wirkungsfreie Transitionen aus, indem Transaktionen mit leeren Vor- und Nachbereichen verboten werden:

$\forall(tr_v \in TR)$: $VB_v \cup NB_v \neq \emptyset$

- Die Beschriftungsbedingung IB_B stellt sicher, daß die Netztopologie und die Netzspezifikation durch die Netzbeschriftung aufeinander abgestimmt werden:

$$
\begin{aligned}
& \forall(s_m \in S)\ \forall(t_n \in T)\ \forall(Prä_u \in PRÄ)\ \forall(tr_v \in TR): \ldots \\
& \qquad (bsp(s_m) = Prä_u \wedge btt(t_n) = tr_v) \\
& \rightarrow \quad (s_m \in VB(t_n) \leftrightarrow Prä_u \in EB(tr_v) \;\wedge\; s_m \in NB(t_n) \leftrightarrow Prä_u \in NB(tr_v))
\end{aligned}
$$

$$
\begin{aligned}
& \forall(s_m \in S)\ \forall(Prä_u \in PRÄ): \ldots \\
& \qquad (bsp(s_m) = Prä_u \wedge (Prä_u, FAK_{u.0}) \in PRÄF) \;\rightarrow\; M_0(s_m) = FAK_{u.0}
\end{aligned}
$$

- Die Markierungsbedingung IB_0 gewährleistet die Zulässigkeit der Netzmarkierung:

$$
\begin{aligned}
& \forall(s_m \in S)\ \forall(Prä_u \in PRÄ): \ldots \\
& \qquad (bsp(s_m) = Prä_u \wedge bsk(s_m) = KAP_m \wedge M_0(s_m) = FAK_{u.0}) \\
& \rightarrow \quad 0 \leq \#(FAK_{u.0}) \leq KAP_m
\end{aligned}
$$

- Die Gewichtungsbedingung IB_G verhindert die Existenz strukturell toter Transitionen:

$\forall(s_m \in S)\ \forall(t_n \in T)$: $\#(bfm(t_n, s_m)) \leq bsk(s_m) \geq \#(bfm(s_m, t_n))$

Erläuterungen und Ergänzungen zur Definition Synthetischer Netze:

a) Synthetische Netze können auch kurz als Netze bezeichnet werden, wenn aus dem Kontext ersichtlich ist, daß Synthetische Netze gemeint sind. In dieser Arbeit werden von nun an bis zum Beginn der Eignungsbeurteilung des Petrinetz-Konzepts nur noch Synthetische Netze betrachtet, sofern nicht ausdrücklich auf einen anderen Netztyp Bezug genommen wird.

b) Die Netztopologie TOP heißt auch topologische Struktur des Synthetischen Netzes SN[6]. Dieses Petrinetz i.e.S. entspricht einem Stelle/Transition-Netz, von dessen Kapazitäts-, Gewichts- und Markierungsfunktionen abstrahiert worden ist.

c) Die Netzspezifikation $SPEC_{MSIG}$ stellt die herausragende Erweiterung Synthetischer Netze gegenüber Stelle/Transition-Netzen dar. Als markenbezogene algebraisch-prädikatenlogische Spezifikation mit zugrundeliegender Markensignatur MSIG bereichert sie Stelle/Transition-Netze, die ausschließlich identische Kopien derselben Basismarke kennen, um das strukturreiche Spektrum sortierter Marken und ihrer - identischen oder verschiedenen - Kopien. Zugleich erweitert die Netzspezifikation die Netztopologie um das formalsprachliche Beschriftungskonzept[7].

d) Die Netzbeschriftung BES und die Netzmarkierung M_0 vermitteln zwischen Netztopologie und Netzspezifikation. Die Beschriftungsfunktionen aus der Netzbeschriftung und die Netzmarkierung bilden Konstrukte aus der Netztopologie auf Konstrukte aus der Netzspezifikation ab. Die Abbildungsergebnisse stellen jeweils beschriftete Netzkomponenten dar[8].

e) Andere Erweiterungen des Petrinetz-Konzepts betonen zumeist entweder nur das algebraische Signaturkonzept oder nur die konventionelle Prädikatenlogik. Zur ersten Alternative gehören z.B. die Arbeiten von Reisig über "Netze mit individuellen Marken"[9] und von Reck über "SIG-Systeme"[10]. Ihre Netzkonzepte unterscheiden sich vom Konzept Synthetischer Netze vor allem dadurch, daß sie die Konstrukte aus der Netztopologie nur mit rein algebraisch definierten Konstrukten beschriften[11]. Hierdurch wird der prädikatenlogische Aspekt vernachlässigt. Die zweite Alternative bilden dagegen die zahlreichen Beiträge zu Prädikat/Transition-Netzen. Durch ihren konventionellen prädikatenlogischen Kalkül lassen sie den Strukturierungsreichtum des algebraischen Signaturkonzepts außer acht. Nur wenige Ausnahmen kombinieren die beiden vorgenannten Alternativen durch Netzbeschriftungen, die - je nach eingenommener Perspektive - auf einer sortierten Prädikatenlogik oder einem prädikatenlogisch erweiterten Signaturkonzept beruhen. Hierzu gehört z.B. das bereits angesprochene SEGRAS-Konzept[12]. Auch die hier vorgestellten Synthetische Netze zeichnen sich dadurch aus, daß in ihrer algebraisch-prädikatenlogischen Netzspezifikation Signaturkonzept und Prädikatenlogik integriert werden.

f) Das topologische Konzept von Petrinetzen i.e.S. einerseits und das algebraisch-prädikatenlogische Signaturkonzept andererseits stellen die beiden konzeptionellen Wurzeln aller Synthetischen Netze dar. Denn jedes Synthetische Netz beruht auf zwei voneinander unabhängigen Fundamenten: seiner Netztopologie TOP und seiner markenbezogenen Spezifikation $SPEC_{MSIG}$ (fundamentale Netzkomponenten). Alle anderen Komponenten aus dem Definitionsschema von Synthetischen Netzen sind auf diesen beiden fundamentalen Netzkomponenten bezogen (derivative Netzkomponenten).

g) Die Prädikatssymbole, Kapazitäten und Markierungen, mit denen die Stellen aus der Netztopologie beschriftet sind, heißen prädikats-, kapazitäts- bzw. markierungsbezogene Stellenanschriften. Entsprechend werden die Transaktionen und die Multimengen atomarer teilevaluierter Formeln, welche die Transitionen bzw. Kanten aus der Netztopologie beschriften, als Transitionen- bzw. Kantenanschriften bezeichnet. Falls $bsp(s_m)=Prä_u$ gilt, heißt das Prädikatssymbol $Prä_u$ ein stellenzugehöriges oder stellenspezifisches Prädikatssymbol. Ebenso wird davon gesprochen, das Prädikatssymbol $Prä_u$ sei der Stelle s_m zugeordnet. Wenn das Prädikatssymbol $Prä_u$ einen natürlichsprachlichen Namen besitzt, kann auch die Stelle s_m mit diesem Namen angesprochen werden[13]. Entsprechend heißt eine Transaktion tr_v einer Transition t_n zugeordnet, wenn $btt(t_n) = tr_v$ gilt. Die Transaktion tr_v wird dann auch als transitionszugehörige oder transitionsspezifische Transaktion bezeichnet. Wiederum läßt sich die Transition t_n mit dem Namen ihrer zugehörigen Transaktion ansprechen, falls es sich um einen natürlichsprachlichen Transaktionsnamen handelt[14].

h) Wegen der jeweils bijektiven Beschriftungen von Transitionen mit Transaktionen und von Stellen mit Prädikatssymbolen können die Bezeichnungen von Konstrukten aus der Netzspezifikation auf Konstrukte der Netztopologie analog[15] übertragen werden[16]:

- ❑ Eine Transition t_n wird genau dann durch das Ausführen der Übergangsprozedur einer Transaktion tr_v geschaltet, wenn $btt(t_n) = tr_v$ gilt. Daher wird die Übergangsprozedur der transitionszugehörigen Transaktion tr_v auch kurz als Schaltprozedur der Transition t_n angesprochen. Ebenso wird das Ausführen der Übergangsoperation einer transitionszugehörigen Transaktion nunmehr vereinfacht als Schalten der Transaktion bezeichnet.
- ❑ Eine Stelle s_m gehört zum Vor-, Informations-, Nach-, Einfluß-, Wirk- oder Operationsbereich der Transition t_n genau dann, wenn gilt:
  - die Stelle s_m ist wegen $bsp(s_m) = Prä_u$ mit dem Prädikatssymbol $Prä_u$ beschriftet;
  - die Transition t_n ist wegen $btt(t_n) = tr_v$ mit der Transaktion tr_v beschriftet;
  - das Prädikatssymbol $Prä_u$ gehört zum Vor-, Informations-, Nach-, Einfluß-, Wirk- bzw. Operationsbereich der Transaktion tr_v.

i) Für jede Transition t_n mit $t_n \in T$ sind aufgrund der voranstehenden Erläuterung folgende transitionsspezifische Stellenmengen ausgezeichnet:

$$VB_n = \{s_m: s_m \in S \wedge bsp(s_m) = Prä_u \wedge btt(t_n) = tr_v \wedge Prä_u \in VB(tr_v)\}$$
$$IB_n = \{s_m: s_m \in S \wedge bsp(s_m) = Prä_u \wedge btt(t_n) = tr_v \wedge Prä_u \in IB(tr_v)\}$$
$$NB_n = \{s_m: s_m \in S \wedge bsp(s_m) = Prä_u \wedge btt(t_n) = tr_v \wedge Prä_u \in NB(tr_v)\}$$
$$EB_n = \{s_m: s_m \in S \wedge bsp(s_m) = Prä_u \wedge btt(t_n) = tr_v \wedge Prä_u \in EB(tr_v)\}$$
$$WB_n = \{s_m: s_m \in S \wedge bsp(s_m) = Prä_u \wedge btt(t_n) = tr_v \wedge Prä_u \in WB(tr_v)\}$$
$$OB_n = \{s_m: s_m \in S \wedge bsp(s_m) = Prä_u \wedge btt(t_n) = tr_v \wedge Prä_u \in OB(tr_v)\}$$

Diese ersten drei dieser Stellenmengen, die aus der Netzspezifikation abgeleitet sind, stimmen mit den transitionsbezogenen Vor- und Nachbereichen, die im Rahmen der Netztopologie als Stellenmengen $VB(t_n)$ bzw. $NB(t_n)$ eingeführt wurden, überein, sofern folgende Zuordnungen beachtet werden:

$$VB(t_n) = VB_n \cup IB_n = EB_n$$

$$NB(t_n) = NB_n$$

Um Mißverständnisse zu vermeiden, lassen sich der spezifikationsdefinierte Vorbereich VB_n und der topologiebedingte Vorbereich $VB(t_n)$ als Vorbereiche der Transition t_n im weiteren bzw. engeren Sinne auseinanderhalten. Beide Vorbereiche unterscheiden sich nur dann, wenn die Transaktion tr_v einen nicht-leeren Informationsbereich $IB(tr_v)$ besitzt. Im Interesse einer klaren Terminologie bevorzugt der Verf. für Synthetische Netze die oben eingeführten, aus der Netzspezifikation abgeleiteten Bezeichnungen für die transitionsspezifischen Stellenmengen. Daher werden fortan Vorbereiche i.e.S. nur als Vorbereiche VB_n und Vorbereiche i.w.S. als Einflußbereiche EB_n angesprochen.

j) Wegen F ((SxT) (TxS)), VB_v $IB_v = \emptyset$ und NB_v $IB_v = \emptyset$ kann jede Kante (kn_x, kn_y) aus der Kantenmenge F eindeutig dem Vor-, Informations- oder Nachbereich einer Transition t_n zugeordnet und entsprechend bezeichnet werden:

- Eingangskante $eka_{n.m}$ der Transition t_n:

  $$(s_m, t_n) = eka_{n.m} :\Leftrightarrow ((s_m, t_n) \in F \wedge bsp(s_m) = Prä_u \wedge btt(t_n) = tr_v \wedge Prä_u \in VB(tr_v))$$

- Informationskante $ika_{n.m}$ der Transition t_n:

  $$(s_m, t_n) = ika_{n.m} :\Leftrightarrow ((s_m, t_n) \in F \wedge bsp(s_m) = Prä_u \wedge btt(t_n) = tr_v \wedge Prä_u \in IB(tr_v))$$

- Ausgangskante $aka_{n.m}$ der Transition t_n:

  $$(t_n, s_m) = aka_{n.m} :\Leftrightarrow ((t_n, s_m) \in F \wedge btt(t_n) = tr_v \wedge bsp(s_m) = Prä_u \wedge Prä_u \in NB(tr_v))$$

Eingangs- und Informationskanten werden unter den Oberbegriff der Einflußkanten subsumiert. Eingangs- und Ausgangskanten gehören zum Oberbegriff der Wirkungskanten.

k) Eine Stelle s_m aus dem Einflußbereich EB_n einer Transition t_n heißt deren Eingangs- bzw. Informationsstelle genau dann, wenn sie mit der Transition durch eine Eingangs- bzw. Informationskante verknüpft ist. Eingangs- und Informationsstellen werden gemeinsam als Einflußstellen ihrer Transition bezeichnet. Eine Stelle s_m aus dem Nachbereich NB_n einer Transition t_n heißt deren Ausgangsstelle; sie ist mit der Transition immer durch eine Ausgangskante verknüpft. Ein- und Ausgangsstellen derselben Transition werden als deren Wirkstellen angesprochen. Die Gesamtheit aller Eingangs-, Informations- und Ausgangsstellen aus dem Operationsbereich OP_n der Transition t_n wird als deren Nachbarschaft bezeichnet. Eine Variable, die in einer Kantenanschrift $MTA_{u.v}$ für eine Kante zwischen einer Stelle s_m mit dem Prädikatssymbol $Prä_u$ und einer Transition t_n mit der Transaktion tr_v enthalten ist, heißt eine Eingangs-, Informations-, Einfluß- oder Ausgangsvariable der Transition genau dann, wenn es sich um die Beschriftung einer Eingangs-, Informations-, Einfluß- bzw. Ausgangskante handelt.

l) Die Netztopologie eines Synthetischen Netzes kann im wesentlichen wie bei Stelle/Transition-Netzen durch einen visualisierten Graphen veranschaulicht werden. Dies wird hier nicht näher ausgeführt, da im anschließenden Kapitel eine spezielle, semi-graphische Darstellungsweise von Synthetischen Netzen ausführlich beschrieben wird.

m) Über die graphische Wiedergabe von Informationskanten, die Elemente der Flußrelation F darstellen, könnte allerdings gestritten werden. Denn die gerichteten Kanten der Flußrelation wurden im Petrinetz-Konzept ursprünglich eingeführt, um in den graphischen Repräsentationen von Stelle/Transition-Netzen schaltbedingte Markenflüsse darzustellen[17)]. Entsprechend wird von einem Flußkonzept der Kanteninterpretation gesprochen. Aus seiner Perspektive werden die Kanten eines Petrinetzes als Flußkanten bezeichnet. Die Markenflüsse können entweder als Markenzuflüsse von den Eingangsstellen einer geschalteten Transition über deren Eingangskanten oder aber als Markenabflüsse auf die Ausgangsstellen einer geschalteten Transition über deren Ausgangskanten erfolgen. Die Kantenrichtung der Ein- und Ausgangskanten von Stelle/Transition-Netzen entspricht daher genau der Richtung aller schaltbedingten Markenflüsse[18)]. Informationskanten zeichnen sich dagegen dadurch aus, daß sie von jedem Markenfluß frei bleiben[19)]. Sie können zwar vermittels ihrer adjazenten Informationsprädikate und ihrer Extensionen das Schaltverhalten von Transitionen beeinflussen, unterliegen hierbei jedoch keinem Markenfluß[20]. Folglich stellt die Relation "F" aus der Netztopologie keine reine Flußrelation mehr dar, sobald mindestens eine Informationskante existiert.

n) Aus diesem scheinbaren Dilemma bestehen zwei Auswege: Entweder wird an der originären Konzeption festgehalten, die Kantenrichtung an die Richtung des Markenflusses in einem Petrinetz zu koppeln. Dann müssen Informationskanten als ungerichtete Kanten eingeführt werden[21]. Oder die ursprüngliche flußbezogene Interpretation der Kantenrichtung wird aufgegeben. Dann läßt sich - im Sinne des nominalistischen Definitionskonzepts - vereinbaren, jede Informationskante (s_m,t_n) mit derselben Kantenrichtung auszuzeichnen, die einer Eingangskante (s_m,t_n) desselben Knotenpaars zukäme[22].

o) Es wird hier die zweite Option realisiert, Informationskanten dieselbe Richtung wie entsprechenden Eingangskanten zu verleihen. Für diese Festlegung sprechen folgende Gründe:

❑ Die simultane Verwendung von gerichteten und ungerichteten Kanten bei der ersten Option würde erhebliche formale Schwierigkeiten bereiten. Denn die graphische Repräsentation der Netztopologie wäre ein hybrider Graph, in dem gerichtete und ungerichtete Graphen miteinander verschmolzen werden müßten. Dem Verf. ist kein Konzept bekannt, solche Hybridgraphen mit formalen Instrumenten der Graphentheorie bearbeiten zu können.

❑ Wenn Informations- und Eingangskanten als gleich gerichtete Einflußkanten behandelt werden, stellt die graphische Repräsentation der Netztopologie weiterhin einen gerichteten Graphen dar. Hierfür stehen leistungsfähige Analyseinstrumente der Graphentheorie zur Verfügung.

❑ Es existiert eine besondere Kategorie von Petrinetzen, die es gestattet, die gerichteten Ausgangs-, Informations- und Ausgangskanten so zu beschriften, daß sich auch für Informationskanten die oben festgelegte Kantenrichtung erklären läßt. Es handelt sich um die Klasse der Numerischen Netze[23]. Bei ihnen besitzt jede Kante weiterhin eine eindeutig definierte Richtung. Alle Kanten werden mit einem Paar aus einer aktivierungsbezogenen Markenmenge und einem schaltbedingten Markenfluß beschriftet. Informationskanten lassen sich in diesen Ansatz problemlos als Spezialfälle von Eingangskanten integrieren[24]. Theoretische Argumente zeichnen diese paarweisen Kantenbeschriftungen als "adäquate" Konstruktion aus[25]. Aufgrund praktischer Überlegungen wird diese Beschriftungsverdopplung hier zwar nicht auf Synthetische Netze übertragen[26]. Sie zeigt aber an, daß die Auszeichnung einer Kantenrichtung für Informationskanten auch theoretisch gerechtfertigt werden kann, wenn sie die Richtung von Eingangskanten erhalten.

Der Preis dieser Festlegung besteht darin, die gerichteten Kanten von Synthetischen Netzen nicht mehr durchgehend als graphische Repräsentationen der Richtung des schaltbedingten Markenflusses interpretieren zu können[27]. Statt dessen fließen entlang einer Informationskante beim Schalten derjenigen Transition, an der die Informationskante endet, überhaupt keine Marken. Informationskanten bleiben daher grundsätzlich frei von jedem Markenfluß[28].

p) Darüber hinaus kann der Verzicht auf eine flußbezogene Kanteninterpretation durch eine neuartige, nunmehr kausal fundierte Kanteninterpretation[29] kompensiert werden: Einerseits zeichnen die gleich gerichteten Eingangs- und Informationskanten einer Transition alle Eingangs- bzw. Informationsprädikate aus, deren Extensionen die Schaltprozedur derjenigen Transaktion beeinflussen, die der betrachteten Transition zugeordnet ist. Die Beeinflussungsmöglichkeiten erstrecken sich auf die Prä-, Inklusions- und Haupttests der Transaktion und deren Bestimmungsgleichungen für die Bindungen der Ausgangsvariablen. Daher wurden die Eingangs- und Informationskanten bereits oben unter den Begriff der Einflußkanten subsumiert. Eingangs- und Informationskanten unterscheiden sich jedoch dadurch, daß nur die Informationsprädikate auf eine reine Beeinflussung der Transaktionsprozedur beschränkt bleiben. Eingangsprädikate werden dagegen durch diese Schaltprozedur uno actu auch beeinflußt, weil ihre Extensionen schaltbedingt verkleinert werden. Schließlich erstrecken sich die Ausgangskanten einer Transition auf alle Ausgangsprädikate, deren Extensionen durch die Schaltprozedur der transitionszugehörigen Transaktion vergrößert werden, ohne selbst diese Schaltprozedur beeinflussen[30] zu können[31]. Folglich gilt: Einflußkanten repräsentieren die Beeinflussung von Schaltprozeduren durch die Extensionen von Einflußprädikaten. Ausgangskanten stellen die beeinflussungsfreie Wirkung von Schaltprozeduren auf die Extensionen von Ausgangsprädikaten dar[32]. Beeinflussung und Wirkung von Schaltprozeduren besitzen kausalen Charakter[33]. Daher werden Einfluß- und Ausgangskanten aus der Netztopologie eines Synthetischen Netzes auch als Kausalkanten bezeichnet. Sie erlauben die Identifizierung folgender kausaler Kategorien[34]:

❑ Reine Beeinflussungen des Schaltens einer Transition, die selbst keinen schaltbedingten Wirkungen unterliegen, werden durch Informationskanten dargestellt.

❑ Reine Wirkungen des Schaltens einer Transition, die dieses Schalten nicht beeinflussen[35], werden von Ausgangskanten repräsentiert.

❑ Kombinierte Beeinflussungen und Wirkungen des Schaltens einer Transition finden ihren Niederschlag in Eingangskanten.

Aufgrund dieser Trichotomie[36] wird die "Fluß"relation F, welche die Gesamtheit aller Kanten in der Netztopologie festlegt, als eine Kausalrelation interpretiert[37]. Entsprechend heißen ihre Elemente Kausalkanten. Daher wird auch von einem kausalen Interpretationskonzept für Synthetische Netze gesprochen.

q) Für die prädikatsbezogenen Stellenanschriften und die Transitionenanschriften von Synthetischen Netzen existieren keine Pendants bei Stelle/Transition-Netzen. Die kapazitätsbezogene Stellenbeschriftung "bsk", die Markierungsfunktion M_0 und die Kantenbeschriftung "bfm" entsprechen dagegen der Kapazitätsfunktion "K", der Markierungsfunktion M_0 bzw. der Gewichtungsfunktion "W" von Stelle/Transition-Netzen. Informationskanten werden nicht nur in Stelle/Transition-Netzen, sondern auch in anderen etablierten Netzklassen nicht berücksichtigt. Dies gilt insbesondere auch für Prädikat/Transition-Netze und andere Höhere Netze. Erst diese Informationskanten erlauben jedoch eine konzeptionell einwandfreie Behandlung von Nebenbedingungen. Dies wird an anderer Stelle detailliert belegt.

q) Die Prädikatssymbole erstrecken sich in ihren Argumenten ausschließlich auf Markensorten. Die prädikatenlogischen Terme, auf welche im Rahmen eines Synthetischen Netzes Prädikatssymbole zur Formierung von prädikatenlogischen Formeln angewendet werden können, stellen daher stets Kopien von sortierten Marken dar. Für die Formelformierung und die Termkonstituierung werden alle Formierungs- bzw. Termbildungsregeln der sortierten Prädikatenlogik erlaubt. Bei den Markenkopien handelt es sich jeweils um Grundterme. Das Ausdrucksspektrum aller zulässigen Formeln Markenkopien ist wegen der Möglichkeiten, auf induktive Weise beliebig komplex zusammengesetzte Formeln und Terme zu bilden, grundsätzlich unbeschränkt. Es kann in Richtung beliebiger Komplexität ausgedehnt werden.

r) Prädikatssymbole und Marken sowie die daraus abgeleiteten Formeln und Markenkopien stellen die konstitutiven, zueinander komplementären Komponenten des prädikatenlogischen Aspekts Synthetischer Netze dar.

s) Aus algebraischer Perspektive besteht eine analoge Komplementarität zwischen Operationssymbolen und Markensorten sowie zwischen ihren Derivaten, dem Operationen bzw. Markenkopien. Allerdings reicht die algebraische Fundierung tiefer. Denn Operationssymbole können sich nicht nur auf Markensorten, sondern ebenso auf Attributsorten erstrecken. Da die Attributsorten aber dem Zweck dienen, Markensorten als Zielsorten aufzubauen, herrscht auch hier eine Dominanz des Markenkonzepts. Daher wurden die zugrundeliegenden MSIG-Signaturen als Markensignaturen bezeichnet.

t) Analog zur Sortensprache, die im Rahmen des allgemeinen Signaturkonzepts eingeführt wurde, läßt sich mit Hilfe der Sortenmenge SO der Markensignatur MSIG eine kombinierte Attribut- und Markensprache AMS_{MSIG} durch AMS_{MSIG} SO^* definieren. Ihre Worte bilden die Gesamtheit aller zulässigen Argumente von Operations- und Prädikatssymbolen in einem Synthetischen Netz.

u) Für Prädikatssymbole gilt sogar nur die positive Markensprache $PMS_{MSIG} \subseteq \{sort_i: i=Q+1,...,Q+3A+2B+2\}^+$. Denn 0-stellige Prädikatssymbole wurden ausgeschlossen. Hinzu kommt, daß sich alle mindestens 1-stelligen Prädikatssymbole sich nur auf Marken-, nicht aber auf Attributsorten erstrecken dürfen. Mit "wort_marke" als einem beliebigen, nicht-leeren Wort aus Markensorten ist die positive Markensprache eines Synthetischen Netzes bestimmt durch:

$$PMS_{MSIG} = \{\text{wort\_marke}: \exists(u=1,...,U): \text{Prä}_u(\text{wort\_marke}) = ... \\ \text{Prä}_u(\text{sor\_marke}_{s(u.1)} ... \text{sor\_marke}_{(u.Ku)})\}$$

v) $SMM=\{m_{s.d}: s=0,1,...,S \wedge d=1,...,D_s\}$ ist die sortierte Markenmenge oder das Markenuniversum eines Synthetischen Netzes. Ihre Elemente $m_{s.d}$ sind Kopien jeweils einer Markenart "s". Alle Kopien derselben Markenart "s" heißen gleichartige Markenkopien. $MAM=\{s: s=0,1, ...,S\}$ stellt die Markenartenmenge des Synthetischen Netzes dar. Diese Konstrukte wurden bereits an früherer Stelle eingeführt.

w) Die Mächtigkeit S+1 der Markenartenmenge kann beliebig groß sein, solange sie mit $S\in \mathcal{N}_+$ endlich bleibt. Als Markenarten werden durch Erzeugungs- und Strukturierungsoperationen alle Markenarten definiert, deren Kopien als Argumente von prädikatenlogischen Formeln grundsätzlich zulässig sein sollen. Ein Synthetisches Netz braucht keineswegs für alle Markenarten, die in ihm definiert sind, auch entsprechende Markenkopien enthalten[38)]. Darüber hinaus ist für ein Synthetisches Netz keine Markenkategorie obligatorisch. Es kann, muß aber nicht die Basismarke oder Attribut- oder Kompositmarken umfassen. Daher kann s=0 undefiniert sein, A=0 gelten bzw. B=0 zutreffen, solange $S\in \mathcal{N}_0$ erfüllt bleibt.

x) Die Mächtigkeit U der Prädikatssymbolemenge PRÄ wird durch die Festlegung U=M fixiert. Dadurch werden nur genau so viele Prädikatssymbole in der Netzspezifikation definiert, wie die Netztopologie Stellen enthält[39)]. Dabei wird mittels Stellenbeschriftung jeder Stelle genau ein stellenspezifisches Prädikatssymbol zugeordnet. Eine Vereinfachung der Netzdefinition läßt sich daher erzielen, indem alle wechselseitig zugeordneten Indizes "m" und "u" von Stellen bzw. Prädikatssymbolen von vornherein miteinander identifiziert werden durch:

$$\forall(m\in \{1,...,M\})\ \forall(u\in \{1,...,U\}):\ bsp(s_m) = \text{Prä}_u \rightarrow m=u$$

y) Die Mächtigkeit V der Transaktionenmenge TR liegt zu V=N fest. Es werden nur genau so viele Transaktionen eingeführt, wie die Netztopologie Transitionen enthält[40)]. Dabei wird mittels Transitionenbeschriftung jeder Transition genau eine transitionenspezifische Transaktion zugeordnet und umgekehrt. Die Netzdefinition läßt sich dadurch vereinfachen, daß alle wechselseitig zugeordneten Indizes "m" und "u" von Transitionen bzw. Transaktionen gleichgesetzt werden gemäß:

$$\forall(n\in \{1,...,N\})\ \forall(v\in \{1,...,V\}):\ btt(t_n) = tr_v \rightarrow n=v$$

z) Infolge ihrer eineindeutigen Zuordnungen ist es zwecks vereinfachter Diktion zulässig, die Stellen mit ihren Prädikatssymbolen und die Transitionen mit ihren Transaktionen gleichzusetzen. Daher gelten alle stellen- oder transitionenspezifischen Formulierungen auch in bezug auf Prädikatssymbole bzw. Transaktionen (vice versa).

A) Die Ausgangsmarkierung M_0 ist eine Funktion, die den Stellen eines Netzes jeweils eine Faktenmultimenge zuschreibt[41]. Dabei wird jede Stelle s_m auf eine Multimenge $FAK_{u.0}$ aus Fakten desjenigen Prädikatssymbols abgebildet, das der Stelle s_m durch $bsp(s_m) = Prä_u$ zugeordnet ist: $M_0(s_m) = FAK_{u.0}$. Die Markierungsfunktion M_0 kann als eine Beschriftungsfunktion sui generis aufgefaßt werden, welche jede Stelle s_m mit einer Faktenmenge $FAK_{u.0}$ beschriftet. Die Faktenmengen werden auch als markierungsbezogene Stellenanschriften bezeichnet. Jede Faktenmenge $FAK_{u.0}$ enthält für ihr Prädikatssymbol $Prä_u$ alle Prädikatsvorkommnisse, die im Ausgangszustand des Netzes mit "r=0" gültig sind[42]. Die Faktenmengen $FAK_{u.0}$, die durch die Markierungsfunktion M_0 den Stellen zugeordnet werden, heißen stellenspezifische Ausgangsfaktenmengen.

B) Jedes Fakt $fakt_0(mu_{u.0}, prä_u(m_{s(u.1)},...,m_{s(u.Ku)}))$, das in einer Faktenmenge $FAK_{u.0}$ eines K_u-stelligen Prädikatssymbols $Prä_u$ enthalten ist, entspricht $mu_{u.0}$ identischen K_u-Tupeln $(m_{s(u.1)},...,m_{s(u.Ku)})$, deren Komponenten $m_{s(u.k)}$ mit $k \in \{1,...,K_u\}$ Kopien der sortierten Marken $sor\_marke_{s(u.k)}$ darstellen. Darauf wird später ausführlicher eingegangen. Hier interessiert zunächst nur die grobe Korrespondenz zwischen Fakten und Markenkopien. Aus dieser Perspektive wird ein Synthetisches Netz als markenfrei bezeichnet, wenn seine Ausgangsmarkierung M_0 jeder Stelle s_m mit $M_0(s_m) = \{\}$ die leere Faktenmenge $FAK_{u.0} = \{\}$ zuordnet. Unter der Ausgangsmarkierung befinden sich dann auf jeder Stelle des Netzes keine Markenkopien. Daher wird die Ausgangsmarkierung mit $M_0(s_m) = \{\}$ für alle $s_m \in S$ auch als Nullmarkierung thematisiert. Ein Netz heißt dagegen markenbehaftet, falls seine Ausgangsmarkierung M_0 mindestens einer Stelle s_m eine nicht-leere Faktenmenge $FAK_{u.0} \neq \{\}$ zuordnet. In besonderen Fällen wird auf die Definition einer Ausgangsmarkierung grundsätzlich verzichtet[43]. Dann wird der Deutlichkeit halber von einem unmarkierten Netz gesprochen. Dagegen wird ein Netz als markiert bezeichnet, wenn herausgestellt werden soll, daß seine Ausgangsmarkierung definiert ist.

C) Die Ausgangsmarkierung läßt sich ebenso als ein geordnetes M-Tupel $\underline{M}_0$ darstellen. Seine Komponenten sind jeweils Bilder $M_0(s_m)$ der Markierungsfunktion M_0 für Stellen s_m und nach Maßgabe aszendenter Stellenindizes "m" angeordnet. Dieses M-Tupel $\underline{M}_0$ stellt einen M-stelligen Spaltenvektor aus stellenspezifischen Funktionsbildern $M_0(s_m)$ dar. Für seinen transponierten Zeilenvektor gilt[44]:

$$\underline{M}_0^{tr} = (M_0(s_1),...,M_0(s_M))$$

mit:

$$\underline{M}_0^{tr} \in (fak_0(mul_0(KAF_{u(1)})) \times ... \times fak_0(mul_0(KAF_{u(M)})))$$

Dasselbe M-Tupel kann auch als eine Familie von stellenspezifischen Funktionsbildern $M_0(s_m)$ aufgefaßt werden. Es wird dann als $MF_0 = (M_0(s_m){:}\ m = 1,...,M)$ notiert. Markierungsfunktion M_0, Markierungsvektor $\underline{M}_0$ und Markierungsfamilie MF_0 unterscheiden sich lediglich durch ihre formale Notation. Ihr materieller Gehalt - die Zuordnung von Faktenmengen zu Stellen - unterscheidet sich nicht. Daher werden sie gemeinsam als Ausgangsmarkierung für den originären Netzzustand "r=0" bezeichnet.

D) Markierungsfunktionen, -vektoren und -Familien werden vom Ausgangszustand "r=0" auf jeden beliebigen Netzzustand "r" mit $r \in \mathcal{N}_+$ verallgemeinert. Die verallgemeinerten Markierungsfunktionen, -vektoren und -Familien sind in derselben Weise definiert wie für den Ausgangszustand "r=0". Markierungsfunktionen, -vektoren und -Familien für beliebige Netzzustände "r" mit $r \in \mathcal{N}_+$ werden gemeinsam unter dem Oberbegriff der Netzmarkierung subsumiert. Wenn ein Synthetisches Netz als Netzmodell dient, auf das ein prädikatenlogisches Objektmodell abgebildet wurde, repräsentiert jede Netzmarkierung zugleich genau einen Zustand des Netzmodells. In diesem Fall stellen Netzmarkierungen sowohl Netz- als auch Modellzustände dar. Beide Sprechweisen können wechselseitig ausgetauscht werden je nachdem, ob die netz- bzw. die modellbezogene Argumentationsperspektive dominiert.

E) Falls die Netzmarkierung als Familie MF_r aller stellenspezifischen Markierungen $M_r(s_m)$ und die Faktenmenge FAK_r als Familie aller prädikatspezifischen Faktenmengen $FAK_{u.r}$ dargestellt werden, sind die Netzmarkierung und die Faktenmenge wegen der bijektiven Zuordnung von Stellen und Prädikatssymbolen für jeden Netzzustand "r" identisch:

$$
\begin{aligned}
& (\, MF_r = (M_r(s_m){:}\ m=1,\ldots,M) \ \wedge\ FAK_r = (FAK_{u.r}{:}\ u=1,\ldots,U) \\
& \wedge\ (\forall(s_m \in S)\ \underline{\exists}(Prä_u \in PRÄ){:}\ \ bsp(s_m) = Prä_u)\,) \\
\Rightarrow\ \ & MF_r = FAK_r
\end{aligned}
$$

F) Bei den alternativen Notationen einer Netzmarkierung als Funktion M_r, Vektor $\underline{M}_r$ oder Mengenfamilie MF_r handelt es sich nur um Notationsvariationen ohne materielle Bedeutung. Gleiches gilt für die Faktenmenge FAK_r. Ihre Notationen als Mengenfamilie oder als Vereinigungsmenge unterscheiden sich zwar formal, aber nicht materiell. Daher erfüllen die Netzmarkierungen M_r, $\underline{M}_r$ und MF_r sowie die Faktenmengen FAK_r unabhängig von der jeweils verwendeten Notationsweise stets die Gleichung: $M_r=\underline{M}_r=MF_r=FAK_r$. Insbesondere gilt auch für die Ausgangsmarkierung: $M_r=\underline{M}_r=MF_0=FAK_0$.

G) Wegen $M_r=\underline{M}_r=MF_r=FAK_r$ besteht eine eineindeutige Zuordnung zwischen Netzmarkierungen und Faktenmengen. Dies gilt sowohl in bezug auf ein ganzes Netz als auch im Hinblick auf einzelne Stellen. Daher können markierungs- und faktenmengenbezogene Formulierungen wechselseitig ausgetauscht werden. Infolgedessen bestehen für Synthetische Netze grundsätzlich zwei alternative Ansätze, das Netzverhalten in der Zeit zu betrachten. Einerseits ist es möglich, sich auf das zugrundeliegende, prädikatenlogisch formulierte Objektmodell zu beziehen. Dann wird das Wissen über den jeweils aktuellen Modellzustand "r" auf der Netzebene durch die aktuelle Faktenmultimenge FAK_r mit Bezug auf Prädikatssymbole $Prä_u$ aus der modellkonstituierenden Prädikatssymbolemenge PRÄ ausgedrückt. Andererseits lassen sich ebenso die Netzmarkierungen M_r betrachten. Sie beziehen sich auf die Stellen s_m aus der Stellenmenge S. Da jeder Stelle s_m aus der Netztopologie mit $bsp(s_m)=Prä_u$ genau ein Prädikatssymbol $Prä_u$ aus dem prädikatenlogischen Objektmodell zugeordnet ist sowie $M_r(s_m)=FAK_{u.r}$ für jede Stelle s_m und jedes zugeordnete Prädikatssymbol $Prä_u$ gilt, besitzen die stellenbezogenen Netzmarkierungen $M_r(s_m)$ und die prädikatssymbolbezogenen Faktenmengen $FAK_{u.r}$ stets denselben Informationsgehalt.

H) Die Netzbeschriftung und die Netzmarkierung bedeuten jeweils Zuordnungen von metasprachlichen Ausdrücken aus der Netzspezifikation zu objektsprachlichen Ausdrücken aus der Netztopologie. Sie konstituieren daher eine formale Netzsemantik. Die Ausdruckszuordnungen werden als Beschriftungen bezeichnet. Dabei legen die beschriftenden Ausdrücke aus der Netzspezifikation die formalen Bedeutungen der beschrifteten Ausdrücke aus der Netztopologie fest. So sind z.B. die formalen Bedeutungen von Stelle ihre zugeordneten Prädikatssymbole. Die formalen Bedeutungen von Transitionen bestehen dagegen aus Transaktionen.

I) Obwohl die Netzbeschriftung und Netzmarkierung in ihrer Funktion übereinstimmen, topologische Netzkonstrukte zu beschriften, werden sie nicht zu einer umfassender definierten Netzbeschriftung zusammengefaßt. Denn zwischen beiden besteht ein charakteristischer Unterschied. Die Beschriftungsfunktionen liefern eine statische Netzbeschriftung, die im Zeitablauf nicht verändert werden kann. Die Netzmarkierung bildet dagegen den Ausgangspunkt einer dynamischen

Netzbeschriftung mit zeitlich variablen Faktenmengen. Die Beschriftungsdynamik wird durch die Ausgangsmarkierung M_0 als Ausgangsfaktenmenge, durch das allgemeine Übergangsschema ÜS für den Übergang zwischen zwei aufeinander folgenden Faktenmengen und durch die Transaktionen tr_v als spezielle Übergangsoperationen konstituiert. Die Netzbeschriftung durch variable Faktenmengen kann durch das Ausführen der Übergangsoperation einer Transaktion, die das Übergangsschema ÜS konkretisiert, verändert werden.

J) Die Netztopologie bildet die syntaktische Dimension eines Synthetischen Netzes. Diese Netzsyntax wird durch Netzspezifikation, Netzbeschriftung und Ausgangsmarkierung in formaler Weise interpretiert[45)]. Sie konstituieren den ersten Aspekt der semantischen Netzdimension. Aus der Perspektive des früher dargelegten semiotischen Bezugsrahmens handelt es sich dabei um eine deklarative Netzsemantik. Darüber hinaus liegt eine zeitlich variable Netzsemantik vor, weil die Netzmarkierungen durch Ausführen der Übergangsoperationen von Transaktionen verändert werden können. Die Gesamtheit aller Netzmarkierungen, die durch zulässige Markierungsveränderungen von der Ausgangsmarkierung aus erreicht werden können, werden durch das allgemeine Übergangsschema und die speziellen Transaktionsdefinitionen in der Netzspezifikation festgelegt. Sie konstituieren als zweiten Aspekt der semantischen Netzdimension eine operationale Netzsemantik. Die Gesamtheit aus deklarativer und operationaler Netzsemantik wird auch als formale Semantik von Synthetischen Netzen bezeichnet.

K) Die formale Netzsemantik verleiht jedem Synthetischen Netz drei bemerkenswerte Eigenschaften:

- Synthetische Netze sind *selbstinterpretierend.* Sie interpretieren einen Teil von sich - die syntaktisch definierte Netztopologie - durch einen anderen Teil: durch die Netzbeschriftungen und -markierungen aus der deklarativen Netzsemantik. Die beiden Aspekte des Interpretandum bzw. des Interpretans sind innerhalb desselben, rein formalen Definitionsschemas für Synthetische Netze erklärt.
- Synthetische Netze verhalten sich *selbstmodifizierend.* Ihre zeitlich variablen Netzmarkierungen werden durch die Übergangsoperationen von Transaktionen und durch das Schalten der zugehörigen Transitionen im Rahmen der operationalen Netzsemantik verändert.
- Synthetische Netze lassen sich durch *bipartite gerichtete beschriftete Graphen mit partiell auto-variabler Beschriftung* repräsentieren. Hierbei wird erstens die topologische Netzstruktur als ein bipartiter gerichteter Graph dargestellt. Zweitens bedeuten die prädikats- und kapazitätsbezogene Stellenbeschriftung sowie die Transitionen- und die Kantenbeschriftung konstante Beschriftungen der Knoten und Kanten des netzrepräsentierenden Graphen. Drittens begründet die Ausgangsmarkierung eine variable Beschriftung der stellenartigen Knoten des Graphen mit (Kopien von) Marken. Diese drei graphischen Eigenschaften resultieren aus der deklarativen Netzsemantik. Viertens erlaubt die operationale Netzsemantik eine autonome, netzendogen erklärte Variation der Beschriftung von stellenartigen Netzknoten mit (Kopien von) Marken.

L) Zwischen den Stellenbeschriftungen durch die Netzmarkierung und den Kantenbeschriftungen durch teilevaluierte atomare Formeln besteht eine bemerkenswerte strukturelle Ähnlichkeit. In beiden Fällen handelt es sich um Multimengen, die über demselben stellenspezifischen Prädikatssymbol definiert sind. Die Multimengen unterscheiden sich nur durch ihre konkreten Elemente (2-Tupel) und durch den Umstand, daß sich markierungsbezogene Stellenanschriften auf konstante Formeln beziehen, die Kantenanschriften dagegen auf teilevaluierte Formeln. Diese Strukturähnlichkeit wird besonders deutlich, wenn die Stellen- und Kantenanschriften als Multimengen in binärer Mengennotation mit positiven Multiplizitäten notiert werden. Mit KAF_u und TAF_u als Mengen aller atomaren konstanten bzw. teilevaluierten Formeln, die aus einem Prädikatssymbol formiert werden können, und mit $MTA_{u.v} \in \{MTAV_{u.v}, MTAI_{u.v}, MTAN_{u.v}\}$ gilt hierfür:

$FAK_{u.0} = Mult_{KAFu.0} = M_0(s_m)$

mit: $Mult_{KAFu.0} \in fak_0(mul_0(KAF_u))$

und: $Mult_{KAFu.0} = \{fakt_0(mu_{u.0},(prä_u(ob_1,...,ob_{Ku}))): ...$
$mu_{u.0} \in \mathcal{N}_0 \wedge prä_u(ob_1,...,ob_{Ku}) \in KAF_u\}$

$MTA_{u.v} = Mult_{TAFu.v}$

mit: $Mult_{TAFu.v} \in mul_v(TAF_u)$

und: $Mult_{TAFu.v} = \{(mu_{u.v},prä_u(te_1,...,te_{Ku})): ...$
$mu_{u.v} \in \mathcal{N}_0 \wedge prä_u(te_1,...,te_{Ku}) \in TAF_u\}$

Auf die Option $Mult_{TAFu.v}=\{\}$ wird im Kernkonzept Synthetischer Netze noch nicht zurückgegriffen. Sie wird erst später für die Definition von Nulltestkanten herangezogen.

M) Aus der Faktenmenge $FAK_{u.r}$ läßt sich die Extension $EXT_{u.r}$ des Prädikatssymbols $Prä_u(st_1,...,st_{Ku})$ im aktuellen Netzzustand "r" ableiten. Die Prädikatsextension ist eine Multimenge von Markentupeln $(m_{s(u.1)},...,m_{s(u.Ku)})$, deren Komponenten jeweils Markenkopien $m_{s(u.k)}$ mit $k \in \{1,...,K_u\}$ darstellen. Jedes dieser Markentupel erfüllt das Prädikatssymbol im Netzzustand "r". Folglich ist das atomare Prädikatsvorkommnis $prä_u(m_{s(u.1)},...,m_{s(u.Ku)})$ in diesem Netzzustand gültig. Ein Markentupel $(m_{s(u.1)},...,m_{s(u.Ku)})$ gehört zur Extension $EXT_{u.r}$ des Prädikatssymbols $Prä_u(st_1,...,st_{Ku})$ genau dann, wenn es zur Konstitution eines Fakts $fakt_r(mu_{u.r}, prä_u(m_{s(u.1)},...,m_{s(u.Ku)}))$ aus der Faktenmenge $FAK_{u.r}$ desselben Prädikatssymbols beiträgt. Daher verhalten sich die Bezugnahmen auf Faktenmengen $FAK_{u.r}$ und auf Prädikatsextensionen $EXT_{u.r}$ äquivalent. Beide werden im wesentlichen durch diejenigen Markentupel $(m_{s(u.1)},...,m_{s(u.Ku)})$ geprägt, welche das zugrundeliegende Prädikatssymbol $Prä_u$ im aktuellen Netzzustand "r" erfüllen. Faktenmengen und Prädikatsextensionen unterscheiden sich nur in ihrer formalen Notation: Faktenmengen umschließen die Markentupel durch Prädikatoren "$prä_u(...)$" und Faktoperatoren "$fakt_r(...)$". Prädikatsextensionen erstrecken sich dagegen nur auf die Markentupel. Daher können faktenmengen- und extensionsbezogene Formulierungen stets ineinander transformiert werden.

N) Netzmarkierungen M_r bilden bei Stelle/Transition-Netzen die Stellen s_m unmittelbar auf Anzahlen $M_r(s_m)$ von Kopien der Basismarke ab. Bei Synthetischen Netzen ordnen die Netzmarkierungen M_r den Stellen s_m zunächst Faktenmultimengen $M_r(s_m)=FAK_{u.r}$ zu. Diese Faktenmengen entsprechen ihrerseits Prädikatsextensionen $EXT_{u.r}$, die Multimengen aus Markentupeln $(m_{s(u.1)},...,m_{s(u.Ku)})$ bestehen. Daher bilden die Netzmarkierungen M_r von Synthetischen Netzen deren Stellen mittelbar auf Multimengen aus Markentupeln ab.

O) Für ein Prädikatssymbol können wegen $mu_{u.r} \in \mathcal{N}_+$ mehrere identische Prädikatsvorkommnisse $prä_u(m_{s(u.1)},...,m_{s(u.Ku)})$ gültig sein (multiple Prädikatsgültigkeit).

P) Wenn unter einer Markierung M_r die aktuelle Faktenmenge $FAK_{u.r}$ einer Stelle s_m mit dem Prädikatssymbol $Prä_u$ das Element $fakt_r(mu_{u.r},prä_u(m_{s(u.1)},...,m_{s(u.Ku)}))$ enthält, so kann dieser Sachverhalt auch dadurch ausgedrückt werden, daß unter der Markierung M_r insgesamt $mu_{u.r}$ Kopien des Markentupels $(m_{s(u.1)},...,m_{s(u.Ku)})$ die Stelle s_m belegen und das zugehörige Prädikatssymbol $Prä_u$ erfüllen.

Q) Wenn das Prädikatssymbol prä_u, das einer Stelle s_m zugeordnet ist, unter der Markierung M_r überhaupt nicht erfüllt wird - also eine leere Extension $EXT_{u.r}$ besitzt, degeneriert die Faktenmenge $FAK_{u.r}$ einer Stelle s_m zur leeren Multimenge: $FAK_{u.r}=\{\}$. Für diesen Sachverhalt wird folgende äquivalente Notation vereinbart[46]:

$$\forall(X_1 \in OB_1) \ldots \forall(X_{Ku} \in OB_{Ku})\colon \text{fakt}_{u.r}(0,\text{prä}_u(X_1,\ldots,X_{Ku}))$$

$$:\Leftrightarrow \quad FAK_{u.r} = \{\}$$

Die faktische Formel $\text{fakt}_{u.r}(0,\text{prä}_u(X_1,\ldots,X_{Ku}))$ drückt aus, daß sich unter jeder beliebigen erreichbaren Markierung M_r auf der Stelle s_m genau Null Kopien eines Markentupels $(m_{s(u.1)},\ldots, m_{s(u.Ku)})$ befinden, für das gilt: Einerseits belegt jede Markenkopie $m_{s(u.k)}$ aus dem Tupel mit $k \in \{1,\ldots,K_u\}$ eine korrespondierende Variable X_k aus dem Argument der variablen atomaren Formel prä_u. Andererseits gehört jede variablenbelegende Markenkopie $m_{s(u.k)}$ zum Definitionsbereich OB_k der Variablen X_k. Er enthält genau alle zulässigen Kopien von Marken der Sorte "$\text{sor\_marke}_{s(u.k)}$". Aufgrund der Allquantifizierung der faktischen Formel $\text{fakt}_{u.r}(0,\text{prä}_u(X_1,\ldots, X_{Ku}))$ werden *alle* Kopien eines Markentupels $(m_{s(u.1)},\ldots,m_{s(u.Ku)})$ erfaßt, die überhaupt zulässig gebildet werden können. Daher ist die allquantifizierte faktische Formel genau dann gültig, wenn die Stelle s_m von Null Kopien eines jeden zulässigen Markentupels $(m_{s(u.1)},\ldots,m_{s(u.Ku)})$ belegt wird. Oder mit anderen Worten: Die allquantifizierte faktische Formel ist genau dann gültig, wenn sich auf der Stelle s_m überhaupt keine Kopie eines Markentupels befindet. Dies ist genau dann der Fall, wenn $FAK_u=\{\}$ zutrifft. Folglich ist die oben eingeführte definitorische Äquivalenz gerechtfertigt. Die Notation der allquantifizierten faktischen Formel fällt zwar erheblich aufwendiger aus als die ursprüngliche Formulierung der leeren Faktenmenge $FAK_u=\{\}$ aus[47]. Aber die alternative Notation wird es später zulassen, eine verallgemeinerte Version von Inhibitorkanten zu formulieren. Ebenso läßt sie sich heranziehen, um Netzrepräsentationen für Integritätsbedingungen auf besonders anschauliche Weise abzuleiten.

R) Jede Formel $\text{prä}_u(m_{s(u.1)},\ldots,m_{s(u.Ku)})$ kann unter Vernachlässigung ihres Namens "prä_u" verkürzt als Tupel $<m_{s(u.1)},\ldots,m_{s(u.Ku)}>$ notiert werden[48]. Darüber hinaus ist es möglich, die Markenkopien $m_{s(u.k)}$ mit $k \in \{1,\ldots,K_u\}$ in der Formel $\text{prä}_u(m_{s(u.1)},\ldots,m_{s(u.Ku)})$ als vollständig explizierte Terme $m_{s(u.k)}{}^{ex}$ zu behandeln, im Tupel $<m_{s(u.1)},\ldots,m_{s(u.Ku)}>$ dagegen nicht in ihre Bestandteile aufzulösen. Wegen dieses Explizierungsverzichts wird die letztgenannte Tupeldarstellung auch als implizite Kurznotation bezeichnet. Die eineindeutige Zuordnung zwischen vollständiger Termexplizierung und impliziter Kurznotation wird fortan durch das Symbol "$\approx$" ausgedrückt. Falls es sich bei den Termen $m_{s(u.k)}$ beispielsweise um die Kopien von Attributmarken mit Erzeugungsoperationen $\text{marke}_{j(s(u.k))}$ handelt, gilt für diese Zuordnung:

$$\begin{aligned}
& \text{prä}_u(m_{s(u.1)},\ldots,m_{s(u.Ku)}) \\
& \wedge\ \mu_{s(u.1)} = \text{marke}_{j(s(u.1))}(at_{j(s(u.1)).1},\ldots,at_{j(s(u.1)).Kj(s(u.1))}) \\
& \ldots \\
& \wedge\ \mu_{s(u.Ku)} = \text{marke}_{j(s(u.Ku))}(at_{j(s(u.Ku)).1},\ldots,at_{j(s(u.Ku)).Kj(s(u.Ku))}) \\
:\Leftrightarrow\quad & <m_{s(u.1)},\ldots,m_{s(u.Ku)}> \\
& \approx\ \text{prä}_u(\text{marke}_{j(s(u.1))}(at_{j(s(u.1)).1},\ldots,at_{j(s(u.1)).Kj(s(u.1))}) \\
& \quad ,\ldots, \\
& \quad \text{marke}_{j(s(u.Ku))}(at_{j(s(u.Ku)).1},\ldots,at_{j(s(u.Ku)).Kj(s(u.Ku))})) \\
:\Leftrightarrow\quad & <m_{s(u.1)},\ldots,m_{s(u.Ku)}> \\
& \approx\ \text{prä}_u(m_{s(u.1)}{}^{ex},\ldots,m_{s(u.Ku)}{}^{ex})
\end{aligned}$$

Analog dazu läßt sich auch jedes Fakt $fakt_r(mu_{u.r}, prä_u(m_{s(u.1)},...,m_{s(u.Ku)}))$ vereinfacht als formaler Ausdruck $mu_{u.r} \bullet <m_{s(u.1)},...,m_{s(u.Ku)}>$ darstellen. Dabei wird zusätzlich von der Faktenbezeichnung "$fakt_r$" abstrahiert. Für $mu_{u.r}=1$ braucht der Faktor "$mu_{u.r}$" dem Tupel $<m_{s(u.1)},...,m_{s(u.Ku)}>$ nicht vorangestellt zu werden. Ebenso ist es möglich, jede Multimenge $MTA_{u.v}$ aus atomaren Formelvorkommnissen $prä_u(m_{s(u.1)},...,m_{s(u.Ku)})$ als eine formale Summe zu notieren, deren Summanden die Gestalt $mu_{u.r} \bullet <m_{s(u.1)},...,m_{s(u.Ku)}>$ annehmen. Notationen dieser Art erlauben später, Synthetische Netze auf besonders kompakte Weise zu repräsentieren. Darüber hinaus entsprechen die Klammersymbole "<" und ">" der vorherrschenden Notation von Individuen-Tupeln in konventionell definierten Prädikat/Transition-Netzen[49]. Dadurch wird der Anschluß von Synthetischen Netzen an Prädikat/Transition-Netze in notationeller Hinsicht gewährleistet.

S) Im Standardfall einstelliger Prädikatssymbole $Prä_u(sor\_marke_{s(u.1)})$ dürfen die Markentupel $(m_{s(u.1)})$ vereinfacht als Markenkopien $m_{s(u)}$ behandelt werden. Dann stellt jede Prädikatsextension eine Multimenge aus Markenkopien dar, welche das zugehörige Prädikatssymbol im aktuellen Netzzustand erfüllen. Wenn unter einer Markierung M_r die aktuelle Faktenmenge $FAK_{u.r}$ der Stelle s_m, die zu dem Prädikatssymbol $Prä_u$ gehört, das Element $fakt_r(mu_{u.r}, prä_u(m_{s(u)}))$ enthält, so wird davon gesprochen, daß $mu_{u.r}$ Markenkopien $m_{s(u)}$ die Stelle s_m unter der Markierung M_r belegen. Die vorgenannten Markenkopien m_s werden zumeist der Einfachheit halber nur als Marken bezeichnet, wenn aus dem Kontext ersichtlich ist, daß es sich um Kopien von Marken handelt.

T) Falls sich ein einstelliges Prädikatssymbol $Prä_u(att\_marke_{s(u.1)})$ auf eine Attributmarke $att\_marke_{s(u.1)}$ erstreckt, können die Notationsvereinfachungen, die in den beiden voranstehenden Erläuterungen eingeführt wurden, kombiniert werden. Ausgangspunkt ist ein Formelvorkommnis $prä_u(m_{s(u)})$ für ein solches Prädikatssymbol, dessen Markenkopie $m_{s(u)}$ die Gleichung $m_{s(u)} = marke_{j(u)}(at_{j(u).1},...,at_{j(u).Kj(u)})$ erfüllt[50]. Dann läßt sich die vereinfachte Schreibweise $<m_{s(u)}>$ für das Formelvorkommnis $prä_u(m_{s(u)})$ unter impliziter Bezugnahme auf den voranstehenden Gleichungszusammenhang durch die aussagekräftigere Notation $<at_{j(u).1},...,at_{j(u).Kj(u)}>$ ersetzen. Entsprechend können Fakten $fakt_r(mu_{u.r}, prä_u(m_{s(u)}))$ und Multimengen aus Formelvorkommnissen $prä_u(m_{s(u)})$ mit der Hilfe der Ausdrücke $mu_{u.r} \bullet <at_{j(u).1},...,at_{j(u).Kj(u)}>$ formuliert werden.

U) Die voranstehend eingeführten Notationsvereinfachungen lassen sich auch auf einstellige Prädikatssymbole $Prä_u(bas\_marke_0)$ anwenden, deren Argumente durch die Basismarke bas_marke_0 festgelegt sind. Jede Kopie der Basismarke läßt sich als "m_0" oder als "$\varnothing$" notieren. Daher gilt für jedes Formelvorkommnis zunächst $prä_u(m_0)$ bzw. $prä_u(\varnothing)$. Dieses Formelvorkommnis kann wieder verkürzt als $<m_0>$ bzw. $<\varnothing>$ notiert werden[51]. Entsprechend lassen sich Fakten $fakt_r(mu_{u.r}, prä_u(m_0))$ und Multimengen aus Formelvorkommnissen $prä_u(m_0)$ vereinfacht als $mu_{u.r} \bullet <m_0>$ darstellen. Analog dazu können die vorgenannten Fakten und Multimengen ebenso als $mu_{u.r} \cdot <\varnothing>$ notiert werden.

V) Die Deklarationen der Transaktion tr_v aus der Transaktionenmenge TR konkretisieren in der Sektion "trans" das allgemeine Übergangsschema ÜS hinsichtlich seiner Vor-, Nach- und Informationsbereiche, seiner Prä-, Haupt- und Posttests sowie seiner Bestimmungsgleichungen in jeweils transaktionsspezifischer Weise. Daher stellt jede Transaktion tr_v mit $v \in \{1,...,V\}$ eine konkrete Übergangsoperation dar, die das Schema ÜS im Sinne der früher beschriebenen operationalen Semantik von prädikatenlogischen Objektmodellierungen erfüllt.

W) Da sich die Übergangsoperationen in jeweils unterschiedlicher Weise ausformulieren lassen, kann neben der Konkretisierung auch eine Individualisierung des Übergangsschemas erfolgen. In welchem Ausmaß dieses Individualisierungspotential ausgeschöpft wird, hängt davon ab, wie stark sich die Deklarationen der Transaktionen tr_v voneinander unterscheiden[52].

X) Die Deklaration einer Transaktion tr_v in der Sektion "trans" kann zusammen mit dem allgemeinen Übergangsschema ÜS, auf das sich alle Transaktionsdeklarationen beziehen, als eine transaktionsspezifische Schaltregel $SR_v=(ÜS,tr_v)$ betrachtet werden. Diese Schaltregel bestimmt, wie eine Referenzfaktenmenge FAK_r durch Ausführen der Übergangsoperation einer Transaktion tr_v unter Beachtung des allgemeinen Übergangsschemas in eine Folgefaktenmenge FAK_f transformiert wird. Aufgrund des Individualisierungspotentials von Transaktionen tr_v lassen sich auch die daraus abgeleiteten Schaltregeln SR_v entsprechend individuell ausgestalten. Damit wird die uniforme Schaltregel von Stelle/Transition-Netzen in Synthetischen Netzen zu einer potentiell[53)] multiformen Schaltregelfamilie SR mit $SR=(SR_v\text{: } v=1,...,V \wedge SR_v=(ÜS,tr_v))$ erweitert.

Y) Jeder Transaktion tr_v wird durch die bijektive Beschriftungsfunktion btt eine Transition t_n mit $tr_v=btt(t_n)$ eineindeutig zugeordnet. Deshalb können die transaktionsspezifischen Schaltregeln SR_v ebenso als transitionsspezifische Schaltregeln SR_n[54)] behandelt werden. Die Deklaration der Transaktion tr_v, die zu einer Transition t_n gehört, spezifiziert daher das Schaltverhalten dieser Transition. Aus diesem Grund läßt sich die Transaktionsdeklaration ebenso als die Schaltvorschrift ihrer zugehörigen Transition ansprechen[55)].

Z) Die transaktionsspezifischen Schaltregeln SR_v sind in der Definition Synthetischer Netze nicht originär enthalten. Sie stellen hier nur ein Derivat aus den Deklarationen der Transaktionen tr_v und aus dem allgemeinen Übergangsschema ÜS dar. Später erlangen die Schaltregeln SR_v aber bei der Implementierung Synthetischer Netze im Rahmen des Softwarepakets PASIPP eine herausragende Rolle. Dort werden sie als PROLOG-Regeln implementiert, die jeweils ein Amalgam aus dem allgemeinen Übergangsschema ÜS mit genau einer Transaktion tr_v darstellen.

α) Für die Formulierung der Prä-, Haupt- und Posttests sowie der Bestimmungsgleichungen einer Transaktion tr_v dienen prädikatenlogische Formeln for_z. Zugleich wird die Transition t_n, der die Transaktion tr_v durch $btt(t_n)=tr_v$ zugeordnet ist, mit diesen Transaktionsformeln for_z mittelbar[56)] beschriftet. Die Formeln sind in der Formelmenge RES_v der Transaktion tr_v entweder als Standardprädikate implizit enthalten oder als besondere Prädikate explizit aufgeführt. Sie lassen sich als Restriktionen oder Gesetze für das Schaltverhalten der jeweils betroffenen Transition auffassen. Denn eine derart (mittelbar) beschriftete Transition t_n kann nur dann geschaltet werden, wenn für die zugehörige Transaktion tr_v eine Variablenbelegungsfunktion vb_c[57)] ausgewählt wird, unter der die Formeln for_z aus den o.a. Tests und Bestimmungsgleichungen erfüllt werden[58)]. Daher werden die Formeln for_z auch als Restriktionsformeln bezeichnet.

β) Die Gesamtheit aller Restriktionsformeln, die an der Formulierung Prä-, Haupt- und Posttests einer Transaktion tr_v teilhaben, lassen sich - in verkürzter Diktion - als die Schaltvoraussetzung der transaktionszugehörigen Transition ansprechen. Ebenso können alle Restriktionsformeln, die Bestimmungsgleichungen derselben Transaktion darstellen, vereinfacht als die Schaltwirkung der transaktionszugehörigen Transition thematisiert werden. In beiden Fällen wird davon abgesehen, daß die Schaltvoraussetzung und -wirkung einer Transition auch von allen anderen Komponenten der Transaktionsdeklaration und des allgemeinen Übergangsschemas abhängen[59)]. Dazu gehört aus der Perspektive der Schaltvoraussetzung z.B. die Überprüfung, ob auf allen Einflußstellen der Transition so viele Markenkopien vorhanden sind, wie von den Gewichten der Einflußkanten der Transition gefordert wird. Zur Schaltwirkung rechnen dagegen auch die Anzahlen von Markenkopien, die aufgrund der Ausgangskantengewichte auf den Ausgangsstellen der Transition abgelegt werden sollen.

χ) Für alle Restriktionsformeln, die Gleichungen darstellen, wird zusätzlich vereinbart: Sie werden mit Hilfe des gewöhnlichen Gleichungsoperators "=" notiert, sofern es sich um einen Prä-, Haupt- und Posttests aus der Schaltvoraussetzung handelt. In den Bestimmungsgleichungen aus der Schaltwirkung wird dagegen der Anweisungsoperator ":=" verwendet. Dadurch wird sichergestellt, daß sich schon aus der Formelnotation erkennen läßt, ob eine Restriktionsformel zu einem Test oder einer Bestimmungsgleichung gehört[60)].

δ) Die Verwendung von Restriktionsformeln für die Spezifizierung von Schaltvoraussetzungen oder -wirkungen ist nicht eindeutig festgelegt. Denn es existieren mehrere Fälle, in denen für dieselbe Transition (mindestens) zwei äquivalente Schaltvorschriften formuliert werden können, die einerseits Restriktionsformeln umfassen und andererseits ohne sie auskommen. Dieser Sachverhalt wird hier nur anhand zweier Beispiele verdeutlicht. Erstens wird die Kopie einer Marke betrachtet, die von der Eingangsstelle einer Transition bei deren Schalten über eine entsprechend gewichtete Eingangskante der Transition abgezogen werden soll. Die Schaltvoraussetzung der Transition legt dafür fest, daß ein Attribut "$attribut_q$" dieser Markenkopie die bestimmte konstante Ausprägung "at_q" annehmen soll. Dies läßt sich auf zwei äquivalente Weise darstellen[61)]: Entweder umfaßt das Gewicht der betroffenen Eingangskante eine Variable "At_q" für die Attributausprägung. Ihr geforderter konstante Wert "at_q" wird dann durch eine Restriktionsformal "$At_q = at_q$" als Haupttestbedingung ausgewiesen[62)]. Oder das Gewicht der Eingangskante enthält anstelle der Variablen "At_q" von vornherein die konstante Attributausprägung "at_q". In diesem Fall erübrigt sich die vorgenannte Haupttestbedingung. Zweitens geht es um die Kopie einer Marke, die auf einer Ausgangsstelle einer Transition bei deren Schalten über eine entsprechend gewichtete Ausgangskante der Transition abgelegt werden soll. Die Schaltwirkung der Transition besteht darin, einem Attribut "$attribut_q$" dieser Markenkopie diejenige Ausprägung "at_q" zuzuweisen, die dasselbe Attribut "$attribut_q$" in einer anderen Markenkopie besitzt, die beim Schalten der Transition über eine ihrer Eingangskanten von einer ihrer Eingangsstellen abgezogen wird. Hierfür existieren wiederum zwei äquivalente Formulierungsalternativen: Entweder werden die Gewichte der betroffenen Ausgangs- und Eingangskante der Transition so festgelegt, daß sie jeweils dieselbe Variable "At_q" für die Attributausprägung umfassen. In diesem Fall sorgt die Technik der Variablenbindung dafür, daß dem Attribut "$attribut_q$" der abgelegten Markenkopie diejenige Ausprägung zugewiesen wird, welche die abgezogene Markenkopie aufgewiesen hat. Oder die Gewichte der Ausgangs- und Eingangskante der Transition erhalten die unterschiedlichen Variablen "$At_{q.neu}$" bzw. "$At_{q.alt}$". Die geforderte Ausprägungsgleichheit kann dann mittels der Bestimmungsgleichung $At_{q.neu} := At_{q.alt}$[63)] ausgedrückt werden[64)].

ε) Für die Prätests aller Prädikatssymbole $Prä_u$ aus dem Informationsbereich $IB(tr_v)$ einer Transaktion tr_v wird als Voreinstellung die immer gültige Tautologie[65)] gewählt: $prätest(FAK_{u.r}) :\Leftrightarrow T$.

ϕ) Haupttests und Bestimmungsgleichungen werden für eine Transaktion tr_v durch transaktionsspezifische Formeln aus der Formelmenge RES_v ausgedrückt[66)]. Bestimmungsgleichungen müssen nicht formuliert sein. Sie entfallen, falls eine Transaktion keine Ausgangsvariablen besitzt oder diese bereits eindeutig mit Einflußvariablen identifiziert worden sind. Falls keine Haupttests expliziert formuliert werden, gelten die impliziten, immer gültigen Voreinstellungen: $haupttest(vb_c(MTA_{u.v})) :\Leftrightarrow T$.

γ) Die Posttests aller Prädikatssymbole $Prä_u$ aus dem Nachbereich $NB(tr_v)$ einer Transaktion tr_v bestehen jeweils aus der Überprüfung, ob die Kapazitäten KAP_m der zugeordneten Stellen s_m unter der hypothetischen Folgemarkierung $FAK_{u.+}$ nicht überschritten werden[67)]:

$$\begin{aligned} & posttest(FAK_{u.+}) \\ :\Leftrightarrow \quad & ((bsp(s_m) = Prä_u \wedge (bsk(s_m) = KAP_m) \rightarrow \#(FAK_{u.+}) \leq KAP_m) \end{aligned}$$

Dabei gibt die Markenkapazität einer Stelle die maximal zulässige Anzahl von Fakten an, die in einem beliebigen Netzzustand in der Faktenmenge des Prädikatssymbols für die jeweils betrachtete Stelle enthalten sein dürfen[68)]. Auf andere mögliche Kapazitätsdefinitionen[69)] wird in dieser Arbeit nicht zurückgegriffen, weil sie für die Bewältigung der späteren Modellierung von Flexiblen Fertigungssystemen keine Rolle spielen.

η) Für die Posttests aller Prädikatssymbole aus dem Vorbereich $VB(tr_v)$ einer Transaktion tr_v wird als Voreinstellung wiederum die immer gültige Tautologie festgelegt: $posttest(FAK_{u,+})$ $:\Leftrightarrow$ T. Dadurch wird die Berücksichtigung von Mindestkapazitäten ausgeschlossen. Sie würden Faktenanzahlen darstellen, die nach dem Schalten einer Transaktion in deren Faktenmenge zurückbleiben müssen. Solche Mindestkapazitäten lassen sich aber bei Bedarf durch eine zusätzliche Beschriftungsfunktion, welche die Stellen auf ihre Mindestkapazitäten abbildet, und eine entsprechende Formulierung der Posttests ohne formale Schwierigkeiten ergänzen[70)].

ι) Die transaktionsspezifische Schaltregel $SR_v=(ÜS,tr_v)$ wird auf die Transaktion tr_v angewendet, indem die Übergangsoperation, die in der Netzspezifikation durch die Transaktion tr_v dargestellt wird, nach der Maßgabe des allgemeinen Übergangsschemas ausgeführt wird. In der Netztopologie ist die ausgeführte Transaktion tr_v durch $tr_v=btt(t_n)$ einer Transition t_n eineindeutig zugeordnet. Daher kann das schaltbedingte Ausführen der Transaktion tr_v ebenso als ein Schalten der korrespondierenden Transition t_n angesehen werden. Fortan wird nicht mehr zwischen dem Ausführen einer Transaktion tr_v und dem Schalten ihrer zugehörigen Transition t_n differenziert. Daher konnte bereits an früherer Stelle auch vom Schalten einer Transaktion gesprochen werden. Ebenso läßt sich auf die Operationsausführung einer Transition t_n Bezug nehmen, wenn hiermit das Ausführen derjenigen Übergangsoperation gemeint ist, die durch die zugehörige Transaktion tr_v dargestellt wird.

φ) Die Variablenbindungsfunktionen oder Variablenbelegungen vb_c[71)] bilden alle Einflußvariablen[72)], die im transaktionsspezifisch konkretisierten Übergangsschema ÜS der Schaltregel SR_v enthalten sind, in sortengerechter Weise auf formale Objekte ab. Dabei kann es sich sowohl um atomare formale Objekte (Konstanten) als auch um beliebig komplex zusammengesetzte formale Objekte handeln. Das Abbildungsergebnis wird für jede betroffene Variable als eine Variablenbindung bezeichnet. Alle formalen Objekte, die für Variablenbindungen in Betracht kommen, stammen aus der Objektmenge $OB=\cup(i=1,...,I:OB_i)$. Sie resultiert aus der Vereinigung aller sortenspezifischen Objektmengen OB_i der Netzspezifikation. Diese Vereinigungsmenge OB ist das prädikatenlogische Universum eines Synthetischen Netzes.

κ) Das Schalten einer Transaktion tr_v unter einer Variablenbelegung vb_c bewirkt nach Maßgabe des Übergangsschemas ÜS, das als Schaltregel SR_v konkretisiert wurde, den Übergang von einer alten Referenz- zu einer neuen Folgemarkierung und den anlogen Übergang zwischen einer korrespondieren Referenz- bzw. Folgefaktenmenge. Dies wird notiert als:

$$M_r\ [tr_v,vb_c\rangle\ M_f$$

$$FAK_r\ [tr_v,vb_c\rangle\ FAK_f$$

Wenn das Schalten einer Transaktion tr_v einen Markierungsübergang bewirkt, wird die zugehörige Variablenbelegung vb_c als ein Übergangs- oder Schaltmodus der Transaktion angesprochen. Der Index "c" des Schaltmodus vb_c stellt einen "Farbindex" dar: Er bezeichnet die "Farbe", in der die Transaktion tr_v geschaltet wird[73)]. Es wird daher auch kurz von der "Schaltfarbe" der Transaktion tr_v gesprochen.

λ) Falls eine Transaktion tr_v einer Transition t_n durch $btt(t_n)=tr_v$ zugeordnet ist und einen Markierungsübergang $M_r[tr_v,vb_c\rangle M_f$ im Schaltmodus vb_c bewirkt, so läßt sich dieser Sachverhalt auch in bezug auf Konstrukte aus der Netztopologie ausdrücken: Das Schalten der Transition t_n mit der Schaltfarbe c_n[74)] bewirkt einen Übergang von der Referenzmarkierung M_r zur Folgemarkierung M_f. Eine analoge Reformulierungsmöglichkeit besteht für den Übergang zwischen zwei aufeinander folgenden Faktenmengen FAK_r und Fak_f. Daher gilt:

$$M_r\ [t_n,c_n\rangle\ M_f \quad :\Leftrightarrow \quad (M_r\ [tr_v,vb_c\rangle\ M_f \wedge btt(t_n)=tr_v)$$

$$FAK_r\ [t_n,c_n\rangle\ FAK_f \quad :\Leftrightarrow \quad (FAK_r\ [tr_v,vb_c\rangle\ FAK_f \wedge btt(t_n)=tr_v)$$

μ) Das Schalten einer Transition ist zwar zunächst als Veränderung der Netzmarkierung und somit als Modifizierung der Faktenmenge definiert. Alle Fakten $fakt_r(mu_{u.r},prä_u(m_{s(u.1)},...,m_{s(u.Ku)})$ aus einer Faktenmenge $FAK_{u.r}$ erstrecken sich aber im wesentlichen auf Markentupel $(m_{s(u.1)},..., m_{s(u.Ku)})$. Daher kann das Schalten einer Transition t_n auch als ein Markenfluß beschrieben werden. Dabei werden Markentupel $(m_{s(u.1)},...,m_{s(u.Ku)})$ von den Eingangsstellen der Transition abgezogen und auf den Ausgangsstellen derselben Transition abgelegt. Dies korrespondiert mit der Ausführung der Übergangsoperation derjenigen Transaktion tr_v, die der Transition t_n durch $btt(t_n)=tr_v$ zugeordnet ist. Dabei werden von der Transaktion Fakten $fakt_r(mu_{u.r},prä_u(m_{s(u.1)},..., m_{s(u.Ku)})$ aus den Referenzfaktenmengen $FAK_{u.r}$ der Prädikatssymbole $Prä_u$ aus ihrem Vorbereich $VB(tr_v)$ entfernt und Fakten $fakt_f(mu_{u.f},prä_u(m_{s(u.1)},...,m_{s(u.Ku)})$ zu den Folgefaktenmengen $FAK_{u.f}$ der Prädikatssymbole $Prä_u$ aus ihrem Nachbereich $NB(tr_v)$ ergänzt.

ν) $MOD_{v.r}=\{vb_c: c=1,...,C_{v.r}\}$ und $MOD_{n.r}=\{c_n: c_n=1,...,C_{n.r}\}$ sind die Mengen aller Schaltmodi vb_c bzw. Schaltfarben c_n, die für eine Transaktion tr_v bzw. ihre zugehörige Transition t_n unter der Netzmarkierung M_r mit $C_{v.r}=C_{n.r}$ zulässig sind. Die Anzahlen $C_{v.r}$ und $C_{n.r}$ zulässiger Schaltmodi bzw. -farben hängen nicht nur von der jeweils betrachteten Transaktion tr_v bzw. Transition t_n ab, sondern ebenso von der aktuellen Netzmarkierung M_r. Falls unter einer Netzmarkierung die Ausführung der Übergangsprozedur der Transaktion unzulässig ist und die zugehörige Transition deshalb nicht geschaltet werden kann, sind die Mengen zulässiger Schaltmodi bzw. -farben jeweils leer. Daher gilt zusätzlich: $MOD_{v.r}=MOD_{n.r}=\{\}$ für $C_{v.r}=C_{n.r}=0$. Mit C_v und C_n werden die Anzahlen der Schaltmodi bzw. -farben notiert, die für eine Transaktion tr_v bzw. eine Transition t_n unter allen erreichbaren Markierungen M_r höchstens zulässig sind. Diese maximalen Anzahlen sind für die hier definierten Synthetischen Netze notwendig endlich[75]. Folglich gilt $C_v \in \mathcal{N}_0$ und $C_n \in \mathcal{N}_0$. Wegen $C_{v.r} \leq C_v$ und $C_{n.r} \leq C_n$ sind daher auch die Anzahlen zulässiger Schaltmodi und -farben durch $C_{v.r} \in \mathcal{N}_0$ bzw. $C_{n.r} \in \mathcal{N}_0$ definiert.

o) Die Menge $MOD_{n.r}$ aller Schaltfarben c_n gibt den Spielraum aller Schaltmöglichkeiten für eine Transition t_n unter einer Markierung M_r an[76]. Im allgemeinen wird das Schaltverhalten der Transition durch die o.a. Restriktionsformeln nicht vollständig determiniert, so daß für den Schaltspielraum $\#(MOD_{n.r}) \geq 2$ gilt. Die Restriktionsformeln prägen daher dem Transitionsschalten in der Regel nur einen nondeterministischen Charakter auf. Sie legen nicht fest, wie die Transition genau schaltet, sondern grenzen nur einen Variationsbereich zulässiger Schaltweisen ein.

π) Eine deterministische Degenerierung dieser nondeterministischen Schaltweise tritt allerdings in zwei Sonderfällen ein. Diese liegen mit $\#(MOD_{n.r}) \leq 1$ vor, falls die Restriktionen derart eng formuliert sind, daß unter der Netzmarkierung M_r nur entweder genau eine oder überhaupt keine Schaltweise einer Transition zulässig ist. Dann ist die Schaltweise dieser Transition bezüglich der Markierung M_r durch die genau eine zulässige Schaltfarbe $c_n=C_{n.r}=1$ mit $MOD_{n.r}=\{c_n\}$ bzw. wegen $MOD_{n.r}=\{\}$ und $C_{n.r}=0$ als Nichtschalten eindeutig determiniert.

θ) Hinsichtlich der nondeterministischen Eingrenzung von Schaltspielräumen für Transitionen durch Restriktionsformeln besteht eine enge Beziehung zwischen dem Petrinetz-Konzept und dem Konzept der restriktionsorientierten Systemmodellierung. Bei dem letztgenannten Modellierungskonzept werden zulässige Modellverhaltensweisen nicht explizit durch Prozeduren definiert. Statt dessen werden alle Verhaltensweisen zugelassen, sofern sie nicht explizit spezifizierte Restriktionen verletzen. Derart konzipierte Systemmodellierungen stellen keineswegs exotische

Spezialfälle dar, sondern spielen im Rahmen der KI-Forschung eine bedeutsame Rolle. Daher wird abermals die Eigenart des Petrinetz-Konzepts deutlich, eine Schnittstelle zu Konzepten der Künstlichen Intelligenz zu eröffnen.

ρ) Eine Transition t_n mit zugeordneter Transaktion tr_v ist unter der Netzmarkierung M_r bezüglich der Schaltfarbe c_n genau dann aktiviert, wenn die Übergangsoperation der Transaktion tr_v unter der Netzmarkierung M_r mit der Variablenbelegung vb_c ausgeführt werden kann[77]. Dieser Sachverhalt wird durch das metasprachliche Prädikat $AKT(t_n,c_n,M_r)$ notiert. Die Transition t_n heißt dann (c_n,M_r)-aktiviert oder c_n-aktiviert unter der Markierung M_r. Das Aktivierungsprädikat wird mit Hilfe des Faktoperators "fak_r", der bereits im Zusammenhang mit dem Konzept der Multimengen eingeführt wurde, festgelegt:

$$AKT(t_n,c_n,M_r)$$

$$:\Leftrightarrow \quad \forall(s_m \in S): \; (M_r(s_m) \geq fak_r(vb_c(bfm(s_m,t_n)))) \qquad {}^{78)}$$

$$\wedge \; (\#(M_r(s_m)) \leq KAP_m + \#(bfm(s_m,t_n)) - \#(bfm(t_n,s_m))) \qquad {}^{79)}$$

σ) Die c_n-Aktivierung einer Transition unter einer Markierung M_r in einem Synthetischen Netz entspricht weitgehend der Aktivierung einer Transition unter einer Markierung in einem Stelle/Transition-Netz. Allerdings kommen zwei Besonderheiten hinzu. Erstens hängt die Transitionsaktivierung von der jeweils ausgewählten Schaltfarbe c_n und der zugrundeliegenden Variablenbelegung vb_c ab. Der Verbund aus Schaltfarbe und Variablenbelegung stellt eine zusätzliche Determinante der Transitionsaktivierung dar. Zugleich bedeutet er einen weiteren Freiheitsgrad bei der Untersuchung oder Gestaltung von Transitionsaktivierungen. Zweitens kann die Aktivierungsbedingung einer Transition nicht mehr - wie bei Stelle/Transition-Netzen - durch Stellenmarkierungen, Kantengewichte und Markenkapazitäten in rein arithmetischer Weise definiert werden. Vielmehr nimmt die Aktivierungsbedingung einer Transition in Synthetischen Netzen eine komplexe Form an. Sie wird einerseits durch das allgemeine Übergangsschema ÜS und durch dessen Konkretisierung vermittels einer transitionszugehörigen Transaktion in implizit-prozeduraler Weise ausgedrückt. Andererseits wird die Aktivierungsbedingung in deklarativer Weise durch das oben angeführte metasprachliche Aktivierungsprädikat $AKT(t_n,c_n,M_r)$ expliziert[80].

τ) Eine Transition t_n heißt M_r-aktiviert oder aktiviert unter einer Markierung M_r, wenn ihre Menge $MOD_{n.r}$ zulässiger Schaltfarben nicht leer ist. Dieser Sachverhalt wird durch das metasprachliche Prädikat $AKT(t_n,M_r)$ ausgedrückt:

$$AKT(t_n,M_r)$$

$$:\Leftrightarrow \quad \forall(c_n \in MOD_{n.r}): AKT(t_n,c_n,M_r)$$

Das Aktivierungsprädikat $AKT(t_n,M_r)$ entspricht - abgesehen von den bereits oben erläuterten Besonderheiten - dem identisch notierten Aktivierungsprädikat $AKT(t_n,M_r)$ für Stelle/Transition-Netze[81].

υ) Eine Transition t_n ist unter der Markierung M_r einfach oder mehrfach aktiviert je nachdem, ob sie genau einen $(C_{n.r}=1)$ bzw. mindestens zwei $(C_{n.r}\geq 2)$ zulässige Schaltfarben c_n für diese Markierung besitzt. Eine mehrfach aktivierte Transition heißt auch multipel aktiviert.

ϖ) Analog zur Vorgehensweise bei Stelle/Transition-Netzen läßt sich definieren, daß unter derselben Markierung M_r zwei Transitionen t_1 und t_2 in entweder nebenläufiger oder aber konfliktionärer Weise c_1- bzw. c_2-aktiviert sind[82]. Allerdings trifft die Analogie nur in dem Ausmaß zu, wie keine Informationsstellen und -kanten berücksichtigt zu werden brauchen. Daher werden vorerst lediglich Transitionen mit jeweils leeren Informationsbereichen betrachtet[83]. Der Übergang von Stelle/Transition zu Synthetischen Netzen bewirkt, daß sich die Aktivierungsprädikate "NEB" bzw. "KON" nunmehr nicht nur auf die betrachtete Netzmarkierung sowie die involvierten Transitionen erstrecken, sondern ebenso auf deren Schaltfarben. Darüber hinaus muß beachtet werden, daß die beiden Transitionen vermittels ihrer Eingangskanten[84] nicht auf dieselbe Marke(n) einer gemeinsamen Eingangsstelle[85] zugreifen dürfen. Denn die Kantenanschriften der Eingangskanten durch Faktenmultimengen $bfm(s_m,t_n)$ und die jeweils ausgewählten Schaltfarben c_n legen Markenmultimengen fest, die beim Schalten der Transitionen von den jeweils betroffenen Eingangsstellen abgezogen werden. Die Transitionen können nur dann unabhängig voneinander schalten, wenn bei diesem Markenabfluß keine Zugriffskonflikte entstehen. Daher reicht es nicht aus zu fordern, daß die Markenmultimengen auf den Eingangsstellen ausreichen, um den Markenabfluß über alle Eingangskanten der beiden Transitionen *der Anzahl* nach abzudecken. Vielmehr muß zusätzlich sichergestellt werden, daß die eingangskantenzugehörigen Faktenmultimengen bei den jeweils ausgewählten Schaltfarben c_n *unter den zugehörigen Variablenbelegungen* vb_c in den Markenmultimengen auf den Eingangsstellen enthalten sind[86]. Daher muß für nebenläufige und konfliktionäre[87] Aktivierung zweier Transitionen gelten[88]:

$$
\begin{array}{ll}
 & NEB((t_1,c_1),(t_2,c_2),M_r) \\
:\Leftrightarrow & AKT(t_1,c_1,M_r) \wedge AKT(t_2,c_2,M_r) \wedge \ldots \\
 & (\forall(s_m \in S):\quad (M_r(s_m) \geq fak_r(vb_{c1}(bfm(s_m,t_1))) + fak_r(vb_{c2}(bfm(s_m,t_2)))) \\
 & \qquad\qquad \wedge\ \ (\#(M_r(s_m)) \leq \ldots \\
 & \qquad\qquad\qquad KAP_m + \#(bfm(s_m,t_1)) + \#(bfm(s_m,t_2)) - \#(bfm(t_1,s_m)) - \#(bfm(t_2,s_m))))
\end{array}
$$

$$
\begin{array}{ll}
 & KON((t_1,c_1),(t_2,c_2),M_r) \\
:\Leftrightarrow & AKT(t_1,c_1,M_r) \wedge AKT(t_2,c_2,M_r) \wedge \ldots \\
 & (\exists(s_m \in S):\quad (\neg (M_r(s_m) \geq fak_r(vb_{c1}(bfm(s_m,t_1))) + fak_r(vb_{c2}(bfm(s_m,t_2))))) \\
 & \qquad\qquad \vee\ \ (\neg (\#(M_r(s_m)) \leq \ldots \\
 & \qquad\qquad\qquad KAP_m + \#(bfm(s_m,t_1)) + \#(bfm(s_m,t_2)) - \#(bfm(t_1,s_m)) - \#(bfm(t_2,s_m)))))
\end{array}
$$

ω) Um Transitionen mit nicht-leeren Informationsbereichen zuzulassen, muß beachtet werden, daß mehrere Transitionen über Informationskanten auf *dieselbe* Marke auf einer gemeinsamen Informationsstelle zugreifen dürfen. Beispielsweise reicht schon *eine* Marke auf einer gemeinsamen Informationsstelle zweier Transitionen aus, um diese beiden Transitionen nebenläufig zu aktivieren. Daher würde es einer fehlerhaften Anzeige ihrer konfliktionären Aktivierung führen, wenn die Beschriftungen von Informationskanten wie in den o.a. Definitionen nebenläufiger und konfliktionärer Aktivierung addiert würden. Allerdings müssen auf einer solchen Informationsstelle mindestens so viele Marken vorhanden sein, damit der Markenzugriff jeder ihrer ausgehenden Informationskanten durch eine entsprechende Variablenbelegung abgedeckt werden kann. Darüber hinaus kann dieselbe Stelle sowohl die Informationsstelle mindestens einer Transition als auch die Eingangsstelle mindestens einer anderen Transition sein. Dann muß die Markenmultimenge auf der Stelle ausreichen, um einerseits den Markenzugriff über alle ausgehenden Informationskanten und zusätzlich den Markenabfluß über alle Ausgangskanten, die keine Informationskanten darstellen, abzudecken[89]. Dabei wird vorausgesetzt, daß eine Marke, die über die Eingangskante einer Transition abfließt, nicht mehr zur Verfügung steht, um über eine Informationskante von einer anderen Transition zur Kenntnis genommen zu werden. Diese Komplikationen werden durch nachfolgende, allgemein gültige Definition der nebenläufigen oder konfliktionären Aktivierung von mehreren Transitionen bewältigt. Sie berücksichtigt die Existenzmöglichkeit von nicht-leeren Informationsbereichen. Darüber hinaus wird sie von vornherein auf eine beliebige Transitionenanzahl $W \in \mathcal{N}_+$ mit $W \geq 2$ ausgedehnt. Dadurch wird erfaßt, daß die paarweise nebenläufige Aktivierung zweier Transitionen verloren gehen kann, wenn das Schalten von mehr als zwei Transitionen unter derselben Markierung versucht wird[90]. Mit dem bereits früher vorgestellten Maximierungsoperator "max" für Multimengenfamilien mit identischen Trägermengen gilt:

$$
\begin{aligned}
& \mathrm{NEB}(\{(t_{n(w)},c_{n(w)}): w=1,\ldots,W \wedge W\geq 2\},M_r) \\
:\Leftrightarrow\quad & \mathrm{AKT}(t_{n(1)},c_{n(1)},M_r) \wedge \ldots \wedge \mathrm{AKT}(t_{n(W)},c_{n(W)},M_r) \wedge \ldots \\
& (\forall (s_m \in S): \ldots \\
& \qquad (M_r(s_m) \geq \ldots \\
& \qquad\qquad \Sigma(w\in\{1,\ldots,W\} \wedge (s_m,t_{n(w)})=eka_{n(w).m}): fak_r(vb_{cn(w)}(bfm(s_m,t_{n(w)}))) \\
& \qquad\qquad + \max\,(fak_r(vb_{cn(w)}(bfm(s_m,t_{n(w)}))): w\in\{1,\ldots,W\} \wedge (s_m,t_{n(w)})=ika_{n(w).m})) \\
& \quad\wedge\quad (\#(M_r(s_m)) \leq \ldots \\
& \qquad\qquad KAP_m \\
& \qquad\qquad + \Sigma(w\in\{1,\ldots,W\} \wedge (s_m,t_{n(w)})=eka_{n(w).m}): \#(bfm(s_m,t_{n(w)})) \\
& \qquad\qquad - \Sigma(w\in\{1,\ldots,W\} \wedge (t_{n(w)},s_m)=aka_{n(w).m}): \#(bfm(t_{n(w)},s_m))))
\end{aligned}
$$

$$
\begin{aligned}
& \mathrm{KON}(\{(t_{n(w)},c_{n(w)}): w=1,\ldots,W \wedge W\geq 2\},M_r) \\
:\Leftrightarrow\quad & \mathrm{AKT}(t_{n(1)},c_{n(1)},M_r) \wedge \ldots \wedge \mathrm{AKT}(t_{n(W)},c_{n(W)},M_r) \wedge \ldots \\
& (\exists (s_m \in S): \ldots \\
& \qquad (\neg(M_r(s_m) \geq \ldots \\
& \qquad\qquad \Sigma(w\in\{1,\ldots,W\} \wedge (s_m,t_{n(w)})=eka_{n(w).m}): fak_r(vb_{cn(w)}(bfm(s_m,t_{n(w)}))) \\
& \qquad\qquad + \max\,(fak_r(vb_{cn(w)}(bfm(s_m,t_{n(w)}))): w\in\{1,\ldots,W\} \wedge (s_m,t_{n(w)})=ika_{n(w).m}))) \\
& \quad\vee\quad (\neg(\#(M_r(s_m)) \leq \ldots \\
& \qquad\qquad KAP_m \\
& \qquad\qquad + \Sigma(w\in\{1,\ldots,W\} \wedge (s_m,t_{n(w)})=eka_{n(w).m}): \#(bfm(s_m,t_{n(w)})) \\
& \qquad\qquad - \Sigma(w\in\{1,\ldots,W\} \wedge (t_{n(w)},s_m)=aka_{n(w).m}): \#(bfm(t_{n(w)},s_m)))))
\end{aligned}
$$

ξ) Ein Schaltschritt SS_a ist eine nicht-leere Menge aus Paaren $(t_{n(w)},c_{n(w)})$ mit $w\in\{1,\ldots W_a\}$ und $W_a \in \mathcal{N}_+$. Jedes Schaltpaar besteht aus einer Transition $t_{n(w)}$ und einer Schaltfarbe $c_{n(w)}$ dieser Transition. W_a ist die Anzahl aller Schaltpaare, die zum Schaltschritt W_a gehören. Zusätzlich wird gefordert, daß jede Transition t_n aus der Transitionenmenge T in einem Schaltschritt nur höchstens einmal vorkommen darf[91]. Daher gilt für jeden Schaltschritt SS_a:

$$
\begin{aligned}
& SS_a = \{(t_{n(w)},c_{n(w)}): \exists (W_a \in \mathcal{N}_+): w=1,\ldots,W_a \wedge \ldots \\
& (\forall (w\in\{1,\ldots,W_a\}): t_{n(w)}\in T \wedge vb_{cn(w)} \in VBM) \wedge \ldots \\
& (\forall (w_1\in\{1,\ldots,W_a\})\ \forall (w_2\in\{1,\ldots,W_a\}): w_1 \neq w_2 \rightarrow t_{n(w1)} \neq t_{n(w2)})\}
\end{aligned}
$$

Ein Schaltschritt SS_a heißt degeneriert oder nicht-degeneriert genau dann, wenn W_a=1 bzw. $W_a \geq 2$ gilt.

ψ) Die Menge SSM aller Schaltschritte, die für ein Synthetisches Netz definiert sind, ergibt sich aus der Transitionenmenge T und aus der Menge VBM aller Variablenbelegungsfunktionen in induktiver Weise. Mit SSM_k wird die Menge aller denkmöglichen Schaltschritte SS_a bezeichnet, die jeweils aus W_a=k Paaren $(t_{n(w)},c_{n(w)})$ bestehen. Da jeder Schaltschritt mindestens ein solches Paar enthalten muß, gilt $k\in \mathcal{N}_+$. Weil in einem Schaltschritt keine Paare $(t_{n(w)},c_{n(w)})$ mit identischen Transitionen vorkommen dürfen und ein Netz mit der Transitionenmenge $T=\{t_n: n=1,...,N\}$ genau N Transitionen umfaßt, kann ein Schaltschritt höchstens k=N Paare $(t_{n(w)},c_{n(w)})$ umfassen. Folglich gilt für die Menge SSM aller denkmöglichen Schaltschritte:

$$SSM_1 = \{\{(t_{n(1)},c_{n(1)})\}: t_{n(1)}\in T \wedge c_{n(1)}\in VBM\}$$

$$\ldots$$

$$\begin{aligned} SSM_k = \{\{(t_{n(w)},c_{n(w)})&: w=1,...,k\}: ... \\ &\{(t_{n(w)},c_{n(w)}): w=1,...,k-1\}\in SSM_{k-1} \\ \cup\quad &\{(t_{n(k)},c_{n(k)})\}\in SSM_1 \wedge t_{n(k)}\neq\{t_{n(w)}: w=1,...,k-1\}\} \end{aligned}$$

$$SSM = \bigcup(k\in\{1,...,N\}): SSM_k$$

ζ) Es wird grundsätzlich die universelle Schaltschritt-Strategie verwendet, sofern keine ausdrücklich abweichenden Festlegungen erfolgen. Daher interessieren unter einer Referenzmarkierung M_r stets *alle* Schaltschritte, die unter dieser Markierung aktiviert sind.

A) Ein degenerierter Schaltschritt SS_a, der aus dem Schaltpaar $(t_{n(1)},c_{n(1)})=(t_n,c_n)$ besteht, heißt genau dann aktiviert unter der Markierung M_r, wenn die Transition t_n unter dieser Markierung c_n-aktiviert ist. Ein nicht-degenerierter Schaltschritt SS_a ist unter der Markierung M_r genau dann aktiviert, falls alle seine Transitionen $t_{n(w)}$ mit $w\in\{1,...,W_a\}$ unter der Markierung M_r nebenläufig aktiviert sind. Die Aktivierung eines Schaltschritts SS_a unter einer Markierung - oder kurz: dessen M_r-Aktivierung - wird daher definiert durch das schaltschrittbezogene Aktivierungsprädikat[92]:

$$\begin{aligned} &AKT(SS_a,M_r) \\ :\Leftrightarrow\quad &(SS_a=\{(t_n,c_n)\} \rightarrow AKT(t_n,c_n,M_r)) \\ &\wedge (SS_a=\{(t_{n(w)},c_{n(w)}): w=1,...,W_a \wedge W_a\geq 2\} \\ &\quad \rightarrow NEB(\{(t_{n(w)},c_{n(w)}): w=1,...,W_a \wedge W_a\geq 2\},M_r))) \end{aligned}$$

B) Falls ein aktivierter Schaltschritt SS_a tatsächlich ausgeführt wird, besteht seine Schaltwirkung darin, die jeweils vorliegende Referenzmarkierung M_r in die Folgemarkierung M_f zu transformieren. Diese Folgemarkierung ist eindeutig determiniert, weil für alle Transitionen eines Schaltschritts die zugehörigen Schaltfarben per definitionem feststehen. Die Schaltwirkung eines Schaltschritts SS_a wird analog zur früheren Notation für das Schalten einer einzelnen Transition t_n mit einer Schaltfarbe c_n dargestellt als[93]:

$$M_r\ [SS_a\rangle\ M_f$$

X) Die Schaltwirkung eines aktivierten degenerierten Schaltschritts stimmt mit der bereits definierten Schaltwirkung einer Transition bezüglich einer Schaltfarbe überein:

$$SS_a = \{(t_n,c_n)\} \wedge AKT(SS_a,M_r)\}$$

$$\Rightarrow \quad M_r\,[SS_a\rangle\,M_f \leftrightarrow M_r\,[t_n,c_n\rangle\,M_f$$

Δ) Die Schaltwirkung eines nicht-degenerierten Schaltschritts ist durch jede beliebige Schaltfolge seiner nebenläufig aktivierten Transitionen mit demselben Schaltergebnis M_f definiert[94]. Daher gilt für jede Permutation der Auflistung von Schaltpaaren $(t_{n(w)},c_{n(w)})$ aus dem Schaltschritt SS_a[95]:

$$SS_a = \{(t_{n(w)},c_{n(w)}):\ w=1,\ldots,W_a \wedge W_a \geq 2\} \ \wedge\ AKT(SS_a,M_r))$$

$$\begin{aligned}
\Rightarrow \quad M_r\,[SS_a\rangle\,M_f \leftrightarrow \ & (\ M_r = M_{r(0)}. \\
& \wedge M_{r(0)}\,[((t_{n(1)},c_{n(1)}))\rangle\,M_{r(1)} \\
& \wedge \ldots \wedge \\
& \wedge M_{r(Wa-1)}\,[((t_{n(Wa)},c_{n(Wa)}))\rangle\,M_{r(Wa)} \\
& \wedge M_{r(Wa)} = M_f\ )
\end{aligned}$$

E) Damit ein nicht-degenerierter Schaltschritt ausgeführt werden kann, müssen alle seine Transitionen nebenläufig aktiviert sein. Daher wird das Ausführen eines solchen Schaltschritts auch als nebenläufiges Schalten seiner Transitionen angesprochen. Ebenso kann vom nebenläufigen Ausführen des Schaltschritts die Rede sein.

Φ) Ein Schaltakt $sa_{r.a.f}$ ist ein Ereignis, bei dem das Ausführen eines Schaltschritts SS_a, der unter einer Markierung M_r aktiviert ist, den Übergang zu einer Folgemarkierung M_f bewirkt:

$$sa_{r.a.f} \ :\Leftrightarrow\ (AKT(SS_a,M_r) \wedge M_r\,[SS_a\rangle\,M_f)$$

Für einen degenerierten Schaltschritt $SS_a = \{(t_{n(1)},c_{n(1)})\} = \{(t_n,c_n)\}$ kann der zugehörige Schaltakt $sa_{r.a.f}$ ebenso als Schaltakt $sa_{r.n.c.f}$ notiert werden. In diesem Fall werden die eine geschaltete Transition t_n und ihre Schaltfarbe c_n im Index der Schaltaktnotation $sa_{r.n.c.f}$ unmittelbar ausgewiesen[96].

Γ) Solange der Schaltschritt SS_a unter der Markierung M_r zwar aktiviert, aber noch nicht ausgeführt ist, heißt er ein potentieller Schaltakt. Wenn dieser Schaltschritt tatsächlich ausgeführt wird, findet der Schaltakt $sa_{r.a.f}$ statt: der potentielle Schaltakt wird realisiert. Hierdurch läßt das Konzept des Schaltakts bescheidene Aspekte der Modal- und der Aktionslogik in das prädikatenlogische Fundament von Synthetischen Netzen einfließen[97].

H) Schaltfolgen SF_L der Länge L mit $L \in \mathcal{N}_0$ und mit Schaltschritten SS_a als Folgengliedern verhalten sich in Synthetischen Netzen genau so wie in Stelle/Transition-Netzen. Daher wird ihre Definition hier nur kurz rekapituliert:

$$SF_L = \begin{cases} () = \varnothing; & \text{für } L=0 \\ (SS_{a(1)}) = (SS_a); & \text{für } L=1 \\ (SS_{a(1)},\ldots SS_{a(L)}); & \text{für } L\geq 2 \end{cases}$$

Die Aktivierung einer Schaltfolge SF_L unter einer Markierung M_r wird durch eine Erweiterung des schaltschrittbezogenen Aktivierungsprädikats abgeleitet. Das Ausführen dieser Schaltfolge SF_L wird durch das sequentielle Ausführen aller schaltfolgenzugehörigen Schaltschritte definiert. Für Nullschaltfolgen und degenerierte Schaltfolgen gelten besondere, aber intuitiv unmittelbar einsichtige Festlegungen.

- Für die Nullschaltfolge $SF_0=\varnothing$ der Länge L=0 gilt:

$$AKT(SF_0,M_r) \;:\Leftrightarrow\; \top$$
$$M_r\,[SF_0\rangle\,M_f \;:\Leftrightarrow\; M_f = M_r$$

- Für degenerierte Schaltfolgen SF_1 der Länge L=1 gilt:

$$AKT(SF_1,M_r) \;:\Leftrightarrow\; AKT(SS_{a(1)},M_r)$$
$$M_r\,[SF_1\rangle\,M_f \;:\Leftrightarrow\; M_r\,[SS_{a(1)}\rangle\,M_f$$

- Für nicht-degenerierte Schaltfolgen SF_L mit L≥2 gilt:

$$AKT(SF_L,M_r) \;:\Leftrightarrow\; AKT(SS_{a(1)},M_r) \wedge M_r = M_{r(0)} \wedge \ldots$$
$$(\forall(l=1,\ldots,L\text{-}1)\text{: } M_{r(l-1)}\,[SS_{a(l)}\rangle\,M_{r(l)} \;\rightarrow\; AKT(SS_{a(l+1)},M_{r(l)}))$$
$$M_r\,[SF_L\rangle\,M_f \;:\Leftrightarrow\; M_{r(0)} = M_r \wedge \ldots$$
$$(\forall(l=1,\ldots,L)\text{: } M_{r(l-1)}\,[SS_{a(l)}\rangle\,M_{r(l)}) \;\wedge\; M_f = M_{r(L)}$$

I) Die Erreichbarkeit von Markierungen ist in Synthetischen Netzen mit der Hilfe von Schaltfolgen SF_L analog wie für Stelle/Transition-Netze definiert[98]. Für die Erreichbarkeitsmenge $RM(M_r)$ einer beliebigen Referenzmarkierung M_r gilt:

$$RM(M_r) \;=\; \{\, M_f\text{: } \underline{M}_f \in (fak_f(mul_f(KAF_1))\mathsf{x}\ldots\mathsf{x}fak_f(mul_f(KAF_U)))$$
$$\wedge\; (\exists(L\in \mathcal{N}_0)\,\exists(SF_L\in SSM^L)\text{: } AKT(SF_L,M_r) \wedge M_r\,[SF_L\rangle\,M_f \,\}$$

Das schließt auch die Erreichbarkeitsmenge $RM(M_0)$ der Ausgangsmarkierung M_0 mit r=0 ein.

ϑ) Der Erreichbarkeitsgraph RG eines Synthetischen Netzes ist durch die voranstehenden Erläuterungen eindeutig festgelegt[99]. Es handelt sich um einen monopartiten, gerichteten und beschrifteten Graphen $RG=(KN_{RG},KA_{RG};bk_{RG})$. Seine Knoten sind die Markierungen M_r, die von der Ausgangsmarkierung M_0 aus durch Schaltfolgen SF_L erreicht werden können (Markierungsknoten). Die Knotenmenge KN_{RG} des Erreichbarkeitsgraphen ist daher mit der Erreichbarkeitsmenge $RM(M_0)$ des zugrundeliegenden Netzes identisch. Die gerichteten Kanten des Erreichbarkeitsgraphen bestehen jeweils aus einem Paar $ka_{r.a.f}=(M_r,M_f)$ erreichbarer Markierungen derart, daß die Referenzmarkierung M_r durch das Ausführen von genau einem Schaltschritt SS_a in die Folgemarkierung M_f transformiert wird (Schaltkanten). KA_{RG} ist die Menge aller solcher Schaltkanten $ka_{r.a.f}$ eines Erreichbarkeitsgraphen. Die Beschriftungsfunktion "bk_{RG}" bildet jedes Element aus der Kantenmenge KA_{RG} des Erreichbarkeitsgraphen auf einen Schaltschritt SS_a aus der Menge SSM aller denkmöglichen Schaltschritte ab. Die Definition der Erreichbarkeitsgraphen von Stelle/Transition-Netzen und die zugehörigen Erläuterungen können ohne größere Schwierigkeiten auf Synthetische Netze übertragen werden. Es ist lediglich zu beachten, daß die größere Ausdrucksmächtigkeit von Synthetischen Netzen erfordert, die Schaltschritte und die Wirkungen von Schaltschrittausführungen zu reformulieren[100]. Daraus folgt für die Komponenten des Erreichbarkeitsgraphen $RG=(KN_{RG},KA_{RG};bk_{RG})$ eines Synthetischen Netzes:

$$KN_{RG} = \{ M_f: \underline{M}_f \in (fak_f(mul_f(KAF_1))x...xfak_f(mul_f(KAF_U))) \wedge (\exists(L \in \mathcal{N}_0)\ \exists(SF_L \in SSM^L): AKT(SF_L,M_0) \wedge M_0\,[SF_L\rangle M_f \}$$

$$KA_{RG} = \{ ka_{r.a.f}: ka_{r.a.f} = (M_r,M_f) \wedge M_r \in KN_{RG} \wedge M_f \in KN_{RG} \wedge (\underline{\exists}(SS_a \in SSM): AKT(SS_a,M_r) \wedge M_r\,[SS_a\rangle M_f \}$$

bk_{RG}: $KA_{RG} \rightarrow SSM$

$ka_{r.a.f} \rightarrow bk_{RG}(ka_{r.a.f})$

mit:

$$bk_{RG}(ka_{r.a.f}) = SS_a \iff (ka_{r.a.f} = (M_r,M_f) \wedge AKT(SS_a,M_r) \wedge M_r\,[SS_a\rangle M_f)$$

Jede beschriftete Kante $ka_{r.a.f}=(M_r,M_f)$ mit $bk_{RG}(ka_{r.a.f})=SS_a$ eines Erreichbarkeitsgraphen kann als ein Schaltakt $sa_{r.a.f}$ aufgefaßt werden. Denn jeder Schaltakt $sa_{r.a.f}$ wurde als ein Ereignis definiert, bei dem ein Schaltschritt SS_a, der unter der Referenzmarkierungen M_r aktiviert ist, ausgeführt wird und dadurch den Übergang zu einer Folgemarkierungen M_f bewirkt: $sa_{r.a.f} :\Leftrightarrow (AKT(SS_a,M_r) \wedge M_r\,[SS_a\rangle M_f)$. Daher läßt sich das Paar aus der Kantenmenge KA_{RG} und aus der Beschriftungsfunktion bk_{RG} mit der Menge SAM aller Schaltakte identifizieren, die im zugrundeliegenden Synthetischen Netz ausgeführt werden können: $(KA_{RG},bk_{RG})=SAM$. Wegen $KN_{RG}=RM(M_0)$ kann der Erreichbarkeitsgraph daher auch als $RG=(RM(M_0),SAM)$ notiert werden[101)].

K) Der Erreichbarkeitsgraph eines Synthetischen Netzes kann wie der eines Stelle/Transition-Netzes einen Multigraphen darstellen. In einem Erreichbarkeits-Multigraphen wird mindestens ein Knotenpaar durch mehrere gleichgerichtete Kanten verknüpft. Die Mitglieder eines solchen Kantenbündels unterscheiden sich nur dadurch, daß sie mit verschiedenen Schaltschritten beschriftet sind. Diese Schaltschritte vermögen die gleiche Gestalt anzunehmen, wie sie an früherer Stelle für Stelle/Transition-Netze exemplarisch vorgeführt wurde. Einen zusätzlichen Freiheitsgrad, der zu weiteren Multikanten führen kann, wird durch das Konzept der Restriktionsformeln für die transitionszugehörigen Transaktionen eröffnet[102)]. Gewöhnlich stellen die Erreichbarkeitsgraphen von Synthetischen Netzen aber Monographen dar[103)].

Λ) Hinsichtlich aller weiteren Sachverhalte, die auf der Erreichbarkeit von Netzmarkierungen beruhen, bestehen keine Unterschiede zwischen Synthetischen Netzen und Stelle/Transition-Netzen. Denn diese Aspekte beziehen sich nicht mehr direkt auf das jeweils zugrundegelegte Netz, sondern auf dessen Erreichbarkeitsgraphen, der für alle Netzklassen dieselbe Struktur besitzt. Daher wird bezüglich solcher Sachverhalte auf die früheren Ausführungen zu Stelle/Transition-Netzen verwiesen. Darüber hinaus werden diese Ausführungen später anläßlich der Erreichbarkeitsanalyse von Synthetischen Netzen vertieft.

M) Die statische Struktur[104)] eines Synthetischen Netzes läßt sich in zwei unterschiedlich weit gefaßten Versionen definieren[105)]. Die Netzstatik i.w.S. umgreift alle Komponenten aus dem 5-Tupel SN für die Definition Synthetischer Netze mit Ausnahme der des Übergangsschemas ÜS, das als eine Subkomponente der Netzspezifikation $SPEC_{MSIG}$ definiert ist: $ST_{sta}^*=(TOP, SPEC_{MSIG}\text{-}(ÜS);BES,M_0;IB)$. Die Netzstatik i.e.S. geht aus der Netzstatik i.w.S. dadurch hervor, daß aus ihr die Ausgangsmarkierung M_0, die zugehörige Integritätsbedingung IB_0 und die Transaktionenmenge TR der Netzspezifikation ausgeschlossen werden. Der Verf. folgt, wenn keine abweichenden Festlegungen getroffen werden, dem eng gefaßten Begriff der Netzstatik. Er besitzt den Vorzug, sich mit dem nachfolgend dargelegten Begriff der Netzdynamik inhaltlich nicht zu überschneiden. Die statische Netzstruktur wird dann als ST_{sta}=(TOP,$SPEC_{MSIG}$-(TR,ÜS); BES;IB-{IB_0}) notiert.

N) Die dynamische Struktur[106)] eines Synthetischen Netzes läßt sich auf drei formal verschiedene, aber inhaltlich äquivalente Weisen definieren. Erstens läßt sich - wie bei der Netzstatik - an das 5-Tupel SN für die Definition Synthetischer Netze anknüpfen. Dann besteht die Netzdynamik aus folgenden drei Komponenten:

- dem allgemeinen Übergangsschema ÜS;
- den Konkretisierungen des Übergangsschemas ÜS durch alle Transaktionen tr_v aus der Transaktionenmenge TR;
- der Ausgangsmarkierung M_0=FAK_0.

Diese erste Variante der Netzdynamik wird als $ST_{DYN.1}$=(ÜS,TR,M_0) oder $ST_{DYN.1}$=(ÜS,TR,FAK_0) notiert. Zweitens können die beiden ersten Komponenten der Netzdynamik zu den individualisierten Schaltregeln SR_v zusammengefaßt und der Ausgangsmarkierung gegenübergestellt werden. Daraus resultiert als nächste Variante der Netzdynamik: $ST_{DYN.2}$=((SR_v: v=1,...,V),M_0) bzw. $ST_{DYN.2}$=(SR_v: v=1,...,V),FAK_0). Drittens ist es möglich, die Netzdynamik mit dem Erreichbarkeitsgraphen RG=(KN_{RG},KA_{RG};bk_{RG})=(RM(M_0),SAM) zu identifizieren: $ST_{DYN.3}$=RG=(KN_{RG},KA_{RG};bk_{RG})=(RM(M_0),SAM). Denn die Ausgangsmarkierung M_0 ist im Erreichbarkeitsgraphen unmittelbar als dessen Ausgangs- oder Wurzelknoten enthalten. Die Kombination aus allgemeinem Übergangsschema ÜS und konkretisierenden Transaktionen tr_v bzw. deren Vereinigung zu individualisierten Schaltregeln SR_v wird durch die Konstruktionsvorschrift des Erreichbarkeitsgraphen implizit abgedeckt: Sie bestimmt alle Schaltakte aus der Schaltaktmenge SAM, die im zugrundeliegenden Netz zulässig sind, um die erreichbaren Markierungen M_r mit $M_r \in$ RM(M_0) zu erzeugen. Der Verf. bevorzugt den dritten Definitionsansatz. Erstens bietet der Erreichbarkeitsgraph eine formal geschlossene und homogene Repräsentation der dynamischen Netzstruktur, die sich darüber hinaus durch ihre graphische Anschaulichkeit auszeichnet. Zweitens kann mit Hilfe des Erreichbarkeitsgraphen auch jedes tatsächliche Netzverhalten kompakt und präzise definiert werden.

O) Statische und dynamische Netzstruktur sind durch ihre zeitliche Invarianz gekennzeichnet. Sie gelten für ein vorliegendes Netz unabhängig von dessen schaltbedingten Markierungsveränderungen. Als ein Netzverhalten wird dagegen jede ununterbrochene zeitliche Abfolge von aufeinanderfolgenden Netzmarkierungen und markierungsverändernden Schaltakten verstanden. Jede solche Abfolge stellt einen Schaltprozeß des betrachteten Netzes dar. Dies wurde bereits für Stelle/Transition-Netze dargelegt. Jedem Netzverhalten entspricht genau ein zusammenhängender, unidirektionaler, aus alternierenden Markierungsknoten und Schaltkanten bestehender Schaltweg im Erreichbarkeitsgraphen. Zugleich repräsentiert die Gesamtheit aller Wege im Erreichbarkeitsgraphen das Potential aller zulässigen Verhaltensweisen des zugrundeliegenden Netzes.

Π) Dynamische Netzstruktur und Netzverhaltensweisen ermöglichen es, die Zustände und Verhaltensweisen eines Synthetischen Netzes zu klassifizieren[107)]:

- Ein Netzzustand "r" heißt genau dann möglich[108)], wenn er durch eine Markierung M_r des Synthetischen Netzes dargestellt werden kann.
- Ein möglicher Netzzustand "r" ist genau dann zulässig, wenn er durch eine Markierung M_r des Synthetischen Netzes repräsentiert wird, die zu seiner Erreichbarkeitsmenge RM(M_0) gehört. Einem zulässigen Netzzustand entspricht daher immer genau ein Knoten aus dem Erreichbarkeitsgraphen, der die dynamische Struktur des Synthetischen Netzes offenlegt.

- Ein Netzzustand "r" heißt genau dann unzulässig, wenn er zwar möglich ist, aber durch keine Markierung M_r aus der Erreichbarkeitsmenge $RM(M_0)$ des Synthetischen Netzes repräsentiert wird. Unzulässige Netzzustände finden sich daher im Erreichbarkeitsgraphen des Synthetischen Netzes nicht als Markierungsknoten wieder.
- Eine Netzverhaltensweise heißt genau dann möglich[109], wenn sie sich als eine Abfolge aus möglichen Markierungen M_r des Synthetischen Netzes darstellen läßt, zwischen denen jeweils Schaltschritte SS_a vermitteln[110].
- Eine mögliche Netzverhaltensweise ist genau dann zulässig, wenn sie durch einen Weg $wg_{r.f}$ im Erreichbarkeitsgraphen des Synthetischen Netzes repräsentiert wird.
- Ein Netzverhaltensweise heißt genau dann unzulässig, wenn sie zwar möglich ist, aber durch keinen Weg $wg_{r.f}$ im Erreichbarkeitsgraphen des Synthetischen Netzes repräsentiert wird.

Darüber hinaus drückt der innere Zusammenhang des Erreichbarkeitsgraphen die Gesamtheit aller Beziehungen zwischen zulässigen Netzzuständen und -verhaltensweisen aus. Da der Erreichbarkeitsgraph selbst als dynamische Netzstruktur definiert wurde, gilt ebenso präzise: Die Netzdynamik definiert das Potential aller zulässigen Netzzustände und aller zulässigen Netzverhaltensweisen einschließlich aller ihrer wechselseitigen Abhängigkeiten.

Θ) Die Integritätsbedingungen aus der Menge IB stellen sicher, daß alle konkreten Synthetische Netze, die das allgemeine Definitionsschema SN erfüllen, zusätzlichen Anforderungen an zulässige Netzkonstruktionen gerecht werden[111]. Einige dieser Integritätsbedingungen wurden bereits für Petrinetze i.e.S. und Stelle/Transition-Netze eingeführt, andere werden hier für Synthetische Netze erstmals aufgestellt[112].

P) Die Disjunktheit-, Existenz- und Verknüpftheitsbedingungen werden aus der Definition von Stelle/Transition-Netzen übernommen. Sie wirken so zusammen, daß die beiden minimalen Netztopologien $TOP_{st}=(\{s_m\},\{t_n\},\{(s_m,t_n)\})$ und $TOP_{ts}=(\{s_m\},\{t_n\},\{(t_n,s_m)\})$ die kleinstmöglichen topologischen Strukturen für Synthetische Netze darstellen. Netze aus höchstens einem Knoten oder Netze mit mindestens einem isolierten Knoten sind topologisch unzulässig. Jede konkret bestimmte[113] Netztopologie, welche die Integritätsbedingungen der Disjunktheit, die Existenz und die Verknüpftheit erfüllt, wird auch dem Definitionsschema A_{TSIG} für topologiebezogene TSIG-Algebren gerecht.

Σ) Die Integritätsbedingungen disjunkter Vor- und Informationsbereiche $VB_v \cap IB_v = \emptyset$ sowie ebenso disjunkter Nach- und Informationsbereiche $NB_v \cap IB_v = \emptyset$ werden neu eingeführt. Sie stellen als Informationsbedingung IB_I sicher, daß die Beschriftung der Eingangs-, Einfluß- und Ausgangskanten aus der Netztopologie mit Multimengen $MTAV_{u.v}$, $MTAI_{u.v}$ bzw. $MTAN_{u.v}$ aus der Netzspezifikation als rechtseindeutige Beschriftungsfunktion "bfm" formuliert werden kann[114]. Bei der ursprünglichen Definition von MSIG-Spezifikationen mußte diese Einschränkung noch nicht vorgenommen werden, weil dort die Verknüpfung mit einer Netztopologie überhaupt noch nicht vorgesehen war. Stelle/Transition-Netze enthielten diese Integritätsbedingungen ebensowenig, aber aus dem entgegengesetzten Grund: Sie besitzen zwar eine wohl definierte Netztopologie, aber keine Netzspezifikation.

T) Die Wirkungsbedingung IB_W schließt Transitionen aus, die zwar schalten können, aber durch ihre Schaltakte im Netz keine Markierungsveränderung bewirken. Solche wirkungsfreien Transitionen können prinzipiell auf zwei Weisen zustandekommen. Einerseits kommen isolierte Transitionen in Betracht, die mit überhaupt keiner Stelle verknüpft sind. Diese Option scheidet jedoch für Synthetische Netze - wie auch für alle anderen Petrinetze (i.e.S.) - aufgrund ihrer Verknüpftheitsbedingungen[115] aus. Andererseits kann es sich um Transitionen handeln, die so mit ihren inzidenten Stellen verknüpft sind, daß gilt: Nur die Schaltvoraussetzung der Transitionen wird von der Markierung ihrer inzidenten Stellen beeinflußt. Falls die aktivierte Transition schaltet, verändert sie jedoch die Markierung keiner ihrer inzidenten Stellen. Solche wirkungsfreien Transitionen können in Petrinetzen (i.e.S.) - wie z.B. Stelle/Transition-Netzen oder Prädikat/Transition-Netzen - durchaus konstruiert werden. Es handelt sich dann um Transition, die so in 1-Schleifen mit jeweils gleich gewichteten Kanten eingebettet sind, daß die Transitionen mit keinen anderen als mit den schleifenzugehörigen Stellen benachbart sind. Die aktuelle Markierung der inzidenten Stellen vermag die Schaltvoraussetzungen dieser Transitionen im Sinne von Nebenbedingungen zu beeinflussen. Aber per constructionem sind die Transitionen außerstande, bei ihrem Schalten die Netzmarkierung zu verändern. Derart wirkungsfreie Transitionen verletzten weder die Verknüpftheitsbedingung noch irgendeine andere konstitutive Eigenschaft von Petrinetzen (i.e.S.). Trotzdem werden sie in dieser Arbeit ausgeschlossen. Denn es widerspräche der Charakterisierung von Transitionen als *aktiven* Netzkomponenten wenn ihre Schaltaktivitäten nichts bewirkten. Aus diesem Grunde werden Synthetische Netze um die Wirkungsbedingung IB_W erweitert. Sie fordert, daß jeder Transition eine Transaktion zugeordnet sein muß, deren Vor- und Nachbereich nicht zugleich leer sein dürfen. Da die Vereinigungsmenge von Vor- und Nachbereich den Wirkungsbereich einer Transaktion darstellt, folgt daraus: Es sind nur Transitionen zugelassen, denen Transaktionen mit nicht-leeren Wirkungsbereichen zugeordnet sind. Daher können wirkungsfreie Transitionen in Synthetischen Netzen grundsätzlich nicht vorkommen[116)].

Y) Die Beschriftungsbedingung IB_B ist rein technischer Art. Sie sorgt dafür, daß die Beschriftungsfunktionen auf dem Tupel BES in einer Weise gewählt werden, welche die Netztopologie und die Netzspezifikation aufeinander abstimmt.

ς) Die Gewichtungs- und die Markierungsbedingung IB_G bzw. IB_0 werden aus Stelle/Transition-Netzen übernommen. Lediglich ihre Notation wird an die Spezifika von Synthetischen Netzen angepaßt.

Ω) Die Menge IB umfaßt vorläufig nur diejenigen Integritätsbedingungen, die vorausgesetzt werden müssen, um überhaupt zulässige Synthetische Netze definieren zu können. Daher handelt es sich um *netzkonstitutive* und definitionsbezogene Integritätsbedingungen. Sie gelten auf der Metaebene für alle Synthetischen Netze. Daher sind sie nicht innerhalb eines Netzes ausformuliert, sondern schränken die Gesamtheit aller zulässigen Netzformulierungen ein (netzexogene Integrität). Des weiteren können aber auf der Objektebene für jedes konkrete Netz zusätzliche *netzspezifische* und objektbezogene Integritätsbedingungen aufgestellt werden. Sie konstituieren nicht die Netzdefinition. Statt dessen drücken sie innerhalb eines bereits definierten Netzes Integritätsbedingungen aus, die von einem modellierten Objekt erfüllt werden sollen[117)]. Diese Integritätsbedingungen werden in einem Netz durch spezielle Netzkonstrukte formuliert (netzendogene Integrität). Sie müssen in jedem Fall explizit angegeben werden. Die Netzkonstrukte für die Repräsentation von Integritätsbedingungen übersteigen jedoch das Ausdrucksvermögen des Kernkonzepts Synthetischer Netze. Daher wird erst später auf sie zurückgekommen, nachdem entsprechende Konzepterweiterungen erfolgt sind.

Ξ) Die voranstehend erläuterten Konstrukte sind für die Definition Synthetischer Netze nur insofern obligatorisch, als das allgemein gültige Definitionsschema[118)] für diese Netzklasse betrachtet wird. Dies war oben der Fall. Bei der Definition eines einzelnen konkreten Synthetisches Netzes, das eine Ausprägung dieses abstrakten Definitionsschemas darstellt, erweisen sich dagegen einige der Definitionskonstituenten als fakultativ. Denn jene Konstituenten des Definitionsschemas, die in unveränderter Weise für alle Synthetischen Netze gelten, brauchen nicht explizit aufgeführt werden. Dann muß aber stets implizit unterstellt werden, daß die vorgelegten konkreten Netze die schematische Definition Synthetischer Netze erfüllen. Unter dieser Voraussetzung brauchen in der Definition eines konkreten Synthetischen Netzes nicht explizit aufgeführt zu werden:

- ❑ das allgemeine Übergangsschema ÜS;

- ❏ die Variablenfamilie VAF;
- ❏ die teilevaluierte Termfamilie TTMF(VAF);
- ❏ die Menge VBM aller zulässigen Variablenbelegungen vb_c;
- ❏ die Menge IB der netzkonstitutiven Integritätsbedingungen.

Daher kann auf die Sektionen "ÜS", "Vars" und "terms" in den Spezifikationen konkreter Netze verzichtet werden[119]. Gleiches gilt für die Menge IB im netzdefinierenden 5-Tupel $SN=(TOP,SPEC_{MSIG};BES,M_0;IB)$, falls diese Menge ausschließlich netzkonstituierende, jedoch keine netzspezifischen Integritätsbedingungen enthält. Die jeweils betroffenen fakultativen Konstituenten aus dem Definitionsschema Synthetischer Netze gelten dann als implizit vereinbart[120].

Ψ) Allen später präsentierten konkreten Netzen liegt die Prämisse zugrunde, daß sie die schematische Definition Synthetischer Netze erfüllen. Entsprechend wird dort auf die explizite Anführung der fakultativen Definitionskonstituenten verzichtet.

Anmerkungen zum Kapitel:

1) Vgl. dazu die Unterscheidung zwischen einem Definitionsschema einerseits, das jeweils eine Netzklasse definiert, und allen konkreten Netzen andererseits, die als Ausprägungen dieses Schemas zu der jeweils definierten Netzklasse gehören.

2) Es handelt sich insofern um eine vorläufige Definition, als hier nur das Kernkonzept Synthetischer Netze definiert wird. Es enthält jedoch bereits alle wesentlichen Definitionskonstituenten. Spätere Erweiterungen des Kernkonzepts Synthetischer Netze erfolgen durch spezielle Netzkonstrukte, die großenteils von der Definition des Kernkonzepts bereits abgedeckt sind. Beispielsweise lassen sich Inhibitorkanten ebenso wie alle Erweiterungen um zeitbezogene Netzkonstrukte in die Definition des Kernkonzepts integrieren. Einige wenige Konzepterweiterungen werden auch zu speziellen Definitionsmodifizierungen führen, wie z.B. die Überlagerung von Schaltprioritäten. Hierdurch wird aber die Netzdefinition des Kernkonzepts nur in Detailaspekten variiert, ohne ihre charakteristische Konstitution aus Netztopologie, -beschriftung und -markierung zu verändern. Andere Konzepterweiterungen werden zwar als Optionen erörtert, aber aufgrund konzeptioneller Bedenken nicht in eine überarbeitete Netzdefinition aufgenommen. Dies betrifft vor allem die mögliche Ergänzung stochastischer Einflußgrößen.

3) Infolge Bijektivität wird jeder Stelle genau ein Prädikatssymbol zugeordnet vice versa. Daher gilt:

$$\forall(s_m \in S)\ \exists(\text{Prä}_u \in \text{PRÄ}):\ bsp(s_m) = \text{Prä}_u$$

$$\forall(\text{Prä}_u \in \text{PRÄ})\ \exists(s_m \in S):\ bsp(s_m) = \text{Prä}_u$$

$$(\#(S) = M \wedge \#(\text{PRÄ}) = U) \rightarrow M = U$$

4) Infolge Bijektivität wird jeder Transition genau eine Transaktion zugeordnet vice versa. Daher gilt:

$$\forall(t_n \in S)\ \exists(tr_v \in TR):\quad btt(t_n) = tr_v$$

$$\forall(tr_v \in TR)\ \exists(t_n \in T):\quad btt(t_n) = tr_v$$

$$(\#(T) = N \wedge \#(TR) = V) \rightarrow N = V$$

5) Da nur die Zuordnung von Multimengen aus teilevaluierten atomaren Formeln gestattet ist, kommen Funktionswerte als Kantenanschriften von Synthetischen Netzen nicht in Betracht. Insbesondere werden alle Funktionen ausgeschlossen, die eine Kante - in Abhängigkeit von der aktuellen Bindung mindestens einer Variablen aus dem Funktionsargument - auf unterschiedliche Multimengen aus teilevaluierten atomaren Formeln abbilden würden. In anderen Höheren Netzen, wie z.B. Prädikat/Transition-Netzen (oder auch Gefärbten Netzen), sind solche Funktionswerte als Kantenanschriften hingegen zulässig. Vgl. dazu das Beispiel der Abb. 37, das in einer der voranstehenden Anmerkungen erläutert wurde. Dort enthalten die Höheren Netze ab der Einführung aggregierter Transitionen u.a. auch Funktionswerte als Kantenanschriften.
Allerdings bedeutet der Verzicht auf funktionsartige Kantenanschriften keine gravierende Einschränkung der Ausdrucksmächtigkeit von Synthetischen Netzen. Denn die Funktionswerte der Kantenanschriften lassen sich auf äquivalente Weise durch Variablen ersetzen. Sie werden in zugehörigen Transitionsanschriften durch partiell definierte Funktionen auf genau jene Funktionswerte eingeschränkt, die in den vorgenannten anderen Höheren Netzen durch die Funktionswerte als Kantenanschriften realisiert werden. Im Beispiel der Abb. 37 läßt sich der Funktionswert $zf(x_{1/2})$, mit dem die Kante $(s_{5/6},t_{1/2})$ beschriftet ist, durch eine Variable $y_{1/2}$ äquivalent ersetzen, indem in der zugehörigen Transition $t_{1/2}$ die partiell definierte Funktion zf mit der Funktionsvorschrift zf: $y_{1/2} \rightarrow_{part} zf(y_{1/2}) = <I>+<II>$; falls $x_{1/2}=A$ und $zf(y_{1/2}) = <II>$; $\phi\alpha\lambda\lambda\sigma$ $x_{1/2}=B$ ergänzt wird. Die partielle Definition der Funktionsvorschrift läßt sich z.B. dadurch realisieren, daß zwei gewöhnlich definierte Funktionen zf_A und zf_B in zwei entsprechend definierte Produktionsregeln eingebettet werden: IF $x_{1/2}=A$ THEN zf_A: $y_{1/2} \rightarrow zf(y_{1/2}) = <I>+<II>$ ENDIF und IF $x_{1/2}=B$ THEN zf_B: $y_{1/2} \rightarrow zf(y_{1/2}) = <II>$ ENDIF. (Statt dessen käme auch eine kompaktere Fallunterscheidung in Betracht: CASE $x_{1/2}=A$ THEN zf_A: $y_{1/2} \rightarrow zf(y_{1/2}) = <I>+<II>$ CASE $x_{1/2}=B$ THEN zf_B: $y_{1/2} \rightarrow zf(y_{1/2}) = <II>$ ENDCASE.)
Allerdings führt der Verzicht auf solche funktionsartigen Kantenanschriften oder äquivalente Kombinationen aus variablen Kantenanschriften und funktionsartigen Transitionsanschriften bei Stelle/Transition-Netzen dazu, daß ihre Ausdrucksmächtigkeit gegenüber Höheren Netzen erheblich eingeschränkt ist. Diese Einschränkung spielt vor allem bei der Kompaktifizierung von Netzen eine Rolle. Denn die maximale Kompaktifizierung eines beliebigen Netzes läßt sich nur garantieren, wenn die vorgenannten funktionsartigen Kanten- oder Transitionsanschriften zugelassen sind. Ohne dieses Ausdrucksmittel kann z.B. das kompakte Netz aus der oben vorgestellten Abb. 37 nicht durch Aggregation von Transitionen aus dem vorangehenden Netz der Abb. 34 gewonnen werden. Denn es ist unmöglich,

die beiden Transitionen t_1 und t_2 (t_3 und t_4) zur aggregierten Transition $t_{1/2}$ ($t_{3/4}$) auf eine andere Weise zusammenzufassen, als es durch die Verwendung der funktionsartigen Kantenanschriften geschehen ist.

6) Ebenso kann von der topologischen Netzstruktur oder dem topologischen Netzkern gesprochen werden.

7) Vgl. z.B. RECK (1988), S. 81 ("Anschriften"); REISIG (1989a), S. 11 ("SIG-inscription").

8) Auch die Netzmarkierung kann als eine Netzbeschriftung - allerdings besonderer Art - aufgefaßt werden.

9) Vgl. REISIG (1989a), S. 10ff.

10) Vgl. RECK (1988), S. 81ff.

11) Dabei handelt es sich im wesentlichen um Sorten, Terme und algebraische Restriktionsformeln (Gesetze); vgl. RECK (1988), S. 81; REISIG (1989a), S. 11.

12) Vgl. z.B. KRÄMER (1987a), S. 270ff., insbesondere S. 272f.; vgl. darüber hinaus die Quellen, die zum SEGRAS-Konzept genannt wurden.

13) Diese Namensübertragung wird durch die bijektive Zuordnung zwischen Stellen und Prädikatssymbolen ermöglicht. Beispielsweise läßt sich eine Stelle s_m als Stelle "Pufferverwaltung" ansprechen, wenn ihr durch die Beschriftung $bsp(s_m)$ = Pufferverwaltung ein Prädikatssymbol mit dem Namen "Pufferverwaltung" eineindeutig zugeordnet ist. Dabei stellt die Redeweise 'Stelle "Pufferverwaltung"' eine Verkürzung der vollständigen Formulierung 'Stelle s_m mit dem Prädikatssymbol "Pufferverwaltung"' dar.

14) Diese Namensübertragung wird abermals durch die bijektive Zuordnung zwischen Transitionen und Transaktionen ermöglicht. Beispielsweise kann eine Transition t_n als Transition "pufferreservierung" angesprochen werden, wenn ihr durch die Beschriftung $btt(t_n)$ = pufferreservierung eine Transaktion mit dem Namen "pufferreservierung" eineindeutig zugeordnet ist. Dabei stellt die Redeweise 'Transition "pufferreservierung"' eine Verkürzung der vollständigen Formulierung 'Transition t_n mit der Transaktion "pufferreservierung"' dar.

15) Dabei ergibt sich lediglich eine Komplikation bezüglich des Vorbereichs, der bei Transitionen einmal im engeren und einmal im weiteren Sinne definiert werden muß, um den Anschluß zur Terminologie der Stelle/Transition-Netze zu wahren.

16) Dabei wird die Terminologie, die für Übergangsoperationen eingeführt wurde, für Transaktionen ohne Veränderung übernommen. Denn Transaktionen stellen konkretisierte Übergangsoperationen dar.

17) Diese markenflußbezogene Kanteninterpretation dominiert die Literatur zum Petrinetz-Konzept. Sie wird daher vom Verf. als konventionelle Interpretation bezeichnet.

18) Gleiches gilt auch für Prädikat/Transition-Netze sowie alle weiteren konventionell definierten Netzklassen.

19) Denn ihren adjazenten Informationsstellen sind Informationsprädikate zugeordnet, deren aktuellen Extensionen durch das Schalten der inzidenten Transitionen per constructionem nicht verändert werden kann. Statt dessen können die Extensionen der Informationsprädikate nur die Schaltprozeduren der zugehörigen Transaktionen beeinflussen. Da den aktuellen Prädikatsextensionen die Markierungen der prädikatszugehörigen Stellen entsprechen, bedeutet die Invarianz der Extensionen von Informationsprädikaten, daß die Markierungen ihrer zugehörigen Informationsstellen während des Schaltens inzidenter Transitionen nicht variiert werden können. Die Marken auf den Informationsstellen können zwar das Schaltverhalten der jeweils betroffenen Transitionen beeinflussen, werden jedoch bei deren Schalten weder bewegt noch verändert. Daher ist es unmöglich, daß Informationsstellen einem schaltbedingten Markenab- oder -zufluß unterliegen; q.e.d.

20) Daher eignen sich Informationskanten und ihre adjazenten Informationsstellen hervorragend, um zwei bedeutsame Konstrukte des Petrinetz-Konzepts abzudecken: 1-Schleifen und Inhibitorkanten. Die erstgenannten wurden bereits mehrfach angesprochen.
Obwohl 1-Schleifen und Inhibitorkanten in Synthetischen Netzen gemeinsam durch Informationsstellen und -kanten erfaßt werden, verhalten sich 1-Schleifen und Inhibitorkanten keineswegs äquivalent, sondern vielmehr komplementär. Denn 1-Schleifen binden die Aktivierung einer Transition an das Vorliegen einer ausreichenden Markierung der schleifenzugehörigen Stelle. Statt dessen bedeuten Inhibitorkanten, daß eine Transition mit einer solchen Kante als Eingangskante nur dann aktiviert sein kann, wenn sich auf der Stelle im Kantenursprung überhaupt keine Marke befindet. Da Informationsstellen und -kanten beide komplementären Aspekte erfassen, erweisen sie sich als ausdrucksstarkes Netzkonstrukt. Dies gilt um so mehr, als später aufgezeigt wird, daß die Darstellungsmöglichkeit von Inhibitorkanten das Ausdrucksvermögen von Netzen erheblich ausweitet. Vgl. dazu die Erläuterungen zur TURING-Mächtigkeit von Netzen mit Nulltestfähigkeit.
Im Hinblick auf 1-Schleifen erlauben Informationskanten, eine Unzulänglichkeit anderer Varianten des Petrinetz-Konzepts zu überwinden. Diese Unzulänglichkeit beruht auf dem Zusammenwirken zweier Aspekte, die keineswegs notwendig sind, aber dennoch zahlreiche Ausgestaltungen des Petrinetz-Konzepts beherrschen: Einerseits wird die Schaltwirkung einer Transition zumeist auf den Fluß von Marken bezogen. Dies wurde schon herausgestellt. Ande-

rerseits liegt des öfteren eine unterschwellige Dominanz der Schaltwirkung gegenüber der Schaltvoraussetzung (Aktivierungsbedingung) vor. Sie führt dazu, daß nur solche Schaltvoraussetzungen aufgestellt werden, die sicherstellen, daß das Schalten einer aktivierten Transition keine unzulässige Netzmarkierung hervorzubringen vermag. Die Kombination der beiden vorgenannten Aspekte bedeutet, daß Schaltvoraussetzungen häufig nur auf Markenflüsse bezogen werden. In diesem Rahmen ist es aber nicht möglich, eine Nebenbedingung auszudrücken. Denn sie zeichnet sich gerade dadurch aus, daß die Markierung der involvierten Nebenbedingungsstelle unverändert bleiben soll. Daher besitzen Nebenbedingungen einen grundsätzlich flußfreien Charakter. Solche flußfreien Nebenbedingungen durch Schaltvoraussetzungen zu erfassen, die nur auf Markenflüsse bezogen sind, erzwingt die artifiziellen Konstrukte der 1-Schleifen. Ihre Schwierigkeiten wurden bereits ausführlich dargelegt. Sie lassen sich jedoch durch die *flußfreien* Informationskanten mühelos überwinden: Eine Informationskante gestattet, die aktuelle Markierung einer Informationsstelle zur Kenntnis zu nehmen, ohne die hiervon betroffenen Markenkopien zu bewegen. Dies entspricht exakt der flußfreien Charakteristik einer Nebenbedingung.

21) Solche ungerichteten Informationskanten werden in der Literatur zum Petrinetz-Konzept nur selten verwendet. Eine der seltenen Ausnahmen hiervon findet sich bei OBERWEIS (1988b), S. 301. Allerdings hält OBERWEIS an der konventionellen Auffassung fest, eine ungerichtete Kante entspreche einem Markenfluß, bei dem Marken zunächst von einer Stelle abgezogen und danach wieder unverändert zurückgelegt würden. Diese Konzeption wurde schon mehrfach im Kontext der Behandlung von Nebenbedingungen thematisiert und kritisiert. Der Auffassung von OBERWEIS entspricht es, wenn mitunter ungerichtete Kanten zugelassen werden, um Paare aus entgegengesetzt gerichteten Kanten in 1-Schleifen zu substituieren; vgl. dazu DITTRICH,G. (1989b), S. 3.

22) Die Alternative, Informationskanten wie Ausgangskanten zu behandeln, wird hier nicht weiter verfolgt. Denn aus der später vorgetragenen kausalen Kanteninterpretation folgt, daß Informations- und Eingangskanten vermittels ihres gemeinsamen Beeinflussungscharakters eng zusammenhängen. Eine entsprechende Verwandtschaft zwischen Informations- und Ausgangskanten besteht dagegen nicht. Vielmehr unterscheiden sie sich in kausaler Hinsicht durch ihren beeinflussenden bzw. wirkungsvermittelnden Charakter diametral. Darüber hinaus zeichnen die theoretischen Argumente, die in der nachfolgenden Erläuterung kurz angesprochen werden, Informations- eindeutig als Derivate von Eingangskanten aus.

23) Vgl. zu Numerischen Netzen BILLINGTON (1982), S. 77ff.; BILLINGTON (1983), S. 173ff.

24) Betrachtet wird hier nur der einfache - aber darum auch übersichtliche - Fall von Stelle/Transition-Netzen. Hierauf sind einerseits Numerische Netze bis heute beschränkt geblieben. Andererseits reichen sie aus, um die Kanten der Netztopologien von Synthetischen Netzen zu untersuchen, weil diese Netztopologien lediglich vereinfachte Varianten von Stelle/Transition-Netzen darstellen. Es wird die Eingangskante (s_m,t_n) einer Transition t_n betrachtet, die in einem Stelle/Transition-Netz mit dem Gewicht $W(s_m,t_n)$ gewichtet ist. Aufgrund der Schaltregel für Stelle/Transition-Netze kann die Transition t_n bezüglich der Stelle s_m nur dann aktiviert sein, wenn die aktuelle Markierung $M_r(s_m)$ dieser Stelle mindestens $W(s_m,t_n)$ Marken umfaßt. Falls die Transition tatsächlich unter der Markierung M_r aktiviert ist und auch geschaltet wird, bewirkt sie einen Markenfluß, der von der Stelle s_m genau $W(s_m,t_n)$ Marken abzieht. Die Kombination aus - hier noch numerisch zusammenfallendem - Aktivierungs- und Schaltaspekt wird nun in Numerischen Netzen durch das modifizierte Kantengewicht $W^*(s_m,t_n)=(wa_{m.n},ws_{m.n})$ ausgedrückt. Dabei bezeichnet $wa_{m.n}$ diejenige Markenanzahl, die auf der Eingangsstelle s_m zur Aktivierung der Transition t_n notwendig ist. Mit $ws_{m.n}$ wird dagegen die Markenanzahl bestimmt, die durch das Schalten der Transition von ihrer Eingangsstelle s_m abgezogen wird. Für gewöhnliche Stelle/Transition-Netze gilt immer $wa_{m.n}=ws_{m.n}=W(s_m,t_n)$. In Numerischen Netzen wird diese Beziehung aber zu $wa_{m.n}\geq ws_{m.n}\geq 0$ verallgemeinert. Dies läßt u.a. auch als spezielle "Eingangs"-kanten einer Transition t_n die Kanten (s_m,t_n) mit $W^*(s_m,t_i)=(wa_{m.n},0)$ und $wa_{m.n}>0$ zu. Solche Kanten können die hier interessierenden Informationskanten in der Netztopologie eines Synthetischen Netzes ausdrücken: Auf der Informationsstelle s_m müssen sich mindestens $wa_{m.n}$ Marken befinden, damit die Extension des zugehörigen Informationsprädikats ausreicht, um die Prä-, Inklusions- und Haupttests einer Transaktion zu erfüllen, die der Transition t_n zugeordnet ist. Falls die Transition t_n tatsächlich schaltet, zieht sie von einer Informationsstelle per constructionem keine Marke ab, weil sie die Extensionen ihrer Informationsprädikate unverändert läßt. Genau diesen Sachverhalt drückt die zweite Komponente $ws_{m.n}=0$ des o.a. Kantengewichts $W^*(s_m,t_i)=(wa_{m.n},0)$ eines Numerischen Netzes aus. Also lassen sich alle Informationskanten in Numerischen Netzen als spezielle Eingangskanten mit der Gewichtskomponente $ws_{m.n}=0$ rekonstruieren; q.e.d. Eine analoge Rekonstruktion von Informationskanten als Spezialfälle von Ausgangskanten gelingt in Numerischen Netzen dagegen nicht. Daher ist es gerechtfertigt, Informations- und Eingangskanten in Synthetischen Netzen als gleichgerichtete Einflußkanten zu behandeln. Sie entsprechen den Eingangskanten von Numerischen Netzen, da in letztgenannten spezielle Informationskanten überhaupt nicht berücksichtigt werden.

25) Es liegt außerhalb des theoretischen Rahmens der hier vorgelegten Untersuchungen, diese Argumente präzise entfalten zu können. Aber zumindest mit einigen Schlagworten läßt sich die relevante Argumentation umreißen: Stelle/Transition-Netze lassen sich als spezielle Variante der allgemeinen arithmetischen Konzepts der Vektor-Additions-Systeme auffassen. In solchen Vektor-Additions-Systemen stellen die Kantengewichte $W(s_m,t_n)$ oder $W(t_n,s_m)$ eines Stelle/Transition-Netzes Operationen dar. Die Operationsausführungen verringern oder vergrößern die Werte

der ganzzahligen Komponenten von Vektoren, die jeweils einer Markierung des zugrundeliegenden Stelle/Transition-Netzes entsprechen. Diese Einbettung von Stelle/Transition-Netzen in Vektor-Additions-Systeme gilt allerdings nur so lange, wie keine beschränkten Markenkapazitäten für die Netzstellen beachtet werden müssen. Für den allgemeinen Fall beschränkter Markenkapazitäten muß dagegen von Vektor-Additions-Systemen zu den allgemeiner definierten Vektor-Ersatz-Systemen übergegangen werden. In solchen Vektor-Ersatz-Systemen werden die vektorverändernden Operationen, die wiederum den Kantengewichten $W(s_m,t_n)$ oder $W(t_n,s_m)$ eines zugrundeliegenden Stelle/Transition-Netzes entsprechen, jeweils in zwei Teiloperationen aufgespalten. Dabei korrespondiert die erste Teiloperation mit der Aktivierungsbedingung der Transition t_n bezüglich der Stelle s_m. Die zweite Teiloperation erstreckt sich dagegen auf die Schaltwirkung, welche die Transition t_n auf die Stelle s_m ausübt. Diese Zweiteilung zwischen aktivierungs- und schaltbezogenen Teiloperationen in Vektor-Ersatz-Systemen reflektieren exakt die ebenso aktivierungs- und schaltbezogen aufgespaltenen Kantengewichte in Numerischen Netzen. Da Vektor-Ersatz-Systeme grundsätzlich *alle* Stelle/Transition-Netze abdecken, können aufgrund der vorgenannten strukturellen Entsprechung Numerische Netze als "adäquate" Reformulierung von Stelle/Transition-Netzen betrachtet werden.

26) Erstens wurde das Konzept Numerischer Netze bisher nur für Petrinetze ausformuliert, welche - abgesehen von der neuartigen Definition gespaltener Kantengewichte - den Stelle/Transition-Netzen entsprechen. Daher bleiben sie auf die strukturlose Basismarke beschränkt. Es liegt außerhalb des Erkenntnisziels der hier vorgelegten Untersuchungen, eine Erweiterung von Numerischen Netzen um das größere Ausdruckspotential der strukturierten Marken von Synthetischen Netzen zu versuchen. Dieses Unterfangen erscheint dem Verf. zwar als grundsätzlich möglich. Es würde aber erhebliche Modifizierungen des Softwarepakets PASIPP erfordern, das hier der Implementierung von Netzmodellen zugrundegelegt wird. Dieser Überarbeitungsaufwand ließe sich allenfalls rechtfertigen, wenn sich die Einführung gerichteter Informationskanten nur mit der Hilfe von Numerischen Netzen rechtfertigen ließe. Der Verf. wird aber nachfolgend zeigen, daß dies auch durch eine revidierte, nunmehr kausale Interpretation von Netzkanten möglich ist. Daher braucht die theoretisch interessante, aber praktisch aufwendige Spaltung der Kantengewichte Numerischer Netze in Synthetischen Netzen nicht zu erfolgen, ohne auf gerichtete Informationskanten verzichten zu müssen.

27) Angesichts der zuvor angeführten Vorteile der Vereinbarung, Informationskanten mit der gleichen Kantenrichtung wie Eingangskanten auszustatten, hält der Verf. diese partielle Unanwendbarkeit der flußbezogenen Interpretation der Kantenrichtungen für akzeptabel.

28) Flußfreie Kanten werden im Rahmen des Petrinetz-Konzepts kaum berücksichtigt. Zu den seltenen Ausnahmen gehört OBERQUELLE (1987a), S. 175f. Er läßt einen "gewinn- und verlustfreien Fluß" zu (S. 175), ohne darüber Auskunft zu erteilen, was in diesem Fall überhaupt noch "fließen" soll. Aus der Abb. 3/5 auf S. 176 läßt sich jedoch erschließen, daß OBERQUELLE einen Informations"fluß" meint, bei dem dasjenige Objekt, dessen Attributausprägungen die interessierenden Informationen (Daten) repräsentieren, ihren Aufenthaltsort nicht verändern. Die objektrepräsentierenden Marken bleiben daher auf den Stellen liegen, auf denen die Informationen ihrer Attributausprägungen abgefragt oder modifiziert werden. Die Informationsabfrage bei OBERQUELLE entspricht genau den hier verwendeten Informationskanten. Dagegen wird hinsichtlich der Informationsmodifizierung von OBERQUELLE's Ansatz abgewichen. Denn der Verf. hält es für angebracht, die Marken, deren Attributausprägungen eine zu modifizierende Information ausdrücken, zunächst durch das Schalten der informationsverändernden Transition abzuziehen und nach der Veränderung ihrer Attributausprägungen wieder abzulegen. Auf diese Weise können informatorische Interferenzprobleme bei nebenläufigen Mehrfachzugriffen auf dieselbe Information vermieden werden.

29) Eine ebenso kausale Kanteninterpretation findet sich bei PAGNONI (1990), S. 122: "The arcs represent causal relationships." Dabei bezieht sich die Autorin auf Petrinetze i.e.S. Sie liegen auch den hier eingeführten Synthetischen Netzen zugrunde. Vgl. ebenso - aber in bezug auf Geschehnisnetze - PAGNONI (1990), S. 164. Vgl. darüber hinaus zu weiteren dezidiert kausalen Interpretationen der Kanten aus Petrinetzen VON MARTIAL (1991a), S. 317.

30) Der Beeinflussungsbegriff erstreckt sich hier nur auf seine oben erfolgte Festlegung, daß die Ergebnisse von Prä-, Inklusions- und Haupttests einer Transaktion und ihrer Bestimmungsgleichungen von den jeweils betrachteten Prädikatsextensionen abhängen können. Darüber hinaus können die Extensionen von Ausgangsprädikaten jedoch die Schaltprozedur einer Transaktion vermittels ihrer Posttests hinfällig werden lassen. *Diese* Abhängigkeit des Schaltverhaltens einer Transaktion wird hier nicht unter den Beeinflussungsbegriff subsumiert. Denn sie dient nicht dazu, um das Transaktionsverhalten in einer bestimmten, erwünschten Richtung zu lenken - es zu "beeinflussen". Vielmehr erfüllt diese letztgenannte Abhängigkeit nur den Zweck, die Integrität eines Netzmodells bezüglich der Markenkapazitäten von Stellen zu gewährleisten. Diese integritätswahrende Funktion wird vom Verf. nicht mehr zum Bedeutungsumfang des Beeinflussungsbegriffs gerechnet.

31) Die Einschränkung des letzten Teilsatzes ist erforderlich, weil auch die Extensionen der Prädikatssymbole, die den Eingangsstellen einer Transition zugeordnet sind, durch die Schaltprozedur der transitionszugehörigen Transaktion verändert werden. Diese Extensionen werden schaltbedingt um die Argumente der entfernten Fakten verringert. Daher spielen Eingangsstellen eine Doppelrolle: Einerseits beeinflussen sie durch die Extensionen ihrer Prädikatssymbole die Schaltprozedur der Transaktion; andererseits werden diese Prädikatsextension aber auch durch die-

selbe Schaltprozedur verändert. In dieser Ambivalenz der kausalen Rolle von Eingangsstellen und zugehörigen Eingangsprädikaten liegt eine Besonderheit der kausalen Interpretation von Petrinetzen. Hierauf wird an späterer Stelle zurückgekommen.

32) Falls die Schaltprozedur einer Transaktion sowohl durch die Extension eines Prädikatssymbols beeinflußt wird als sich auch auf diese Extension verändernd auswirkt, liegt eine kombinierte Schaltbeeinflussung und -wirkung bezüglich derselben Prädikatsextension vor. Dieser Kombinationsfall einer beeinflußten Schaltwirkung wird in kausal interpretierten Petrinetzen durch zwei Alternativen erfaßt: Wenn die beeinflußte Schaltwirkung *nur* aus einer Verringerung der Prädikatsextension besteht, dann gehört das betroffene Prädikatssymbol nur zum Vorbereich der schaltenden Transaktion, nicht aber zu deren Nachbereich. Wenn die beeinflußte Schaltwirkung dagegen die Prädikatsextension *sowohl* um Prädikatsargumente verringert *als auch* erweitert, dann zählt das betroffene Prädikatssymbol sowohl zum Vor- als auch zum Nachbereich der schaltenden Transaktion. Daher wurde früher zugelassen, daß diese beiden Transaktionsbereiche keineswegs disjunkt zu sein brauchen. Diese Option wird auch genutzt, um Nebenbedingungen befriedigend darzustellen.
Der dritte Fall, daß die beeinflußte Schaltwirkung *nur* eine Erweiterung, aber keine Verringerung der Prädikatsextension hervorbringt, ist unter den zuvor getroffenen Festlegungen logisch ausgeschlossen. Denn Beeinflussungen von Schaltprozeduren wurden an Einflüsse auf Prä-, Inklusions- oder Haupttests oder auf Variablenbindungen in Bestimmungsgleichungen geknüpft. Solche Beeinflussungen können aufgrund der Konstruktion des Übergangsschemas ÜS nur von Eingangs- oder Informationsprädikaten ausgehen. Die Schaltprozeduren von Transaktionen üben aber keine verändernde Wirkung auf die Extensionen von Informationsprädikaten aus. Daher bleiben für den dritten Fall nur noch Eingangsprädikate übrig. Deren Extensionen werden jedoch von Schaltprozeduren der geschalteten Transaktionen - wiederum aufgrund des Übergangsschemas ÜS - notwendig verringert. Dies widerspricht der Prämisse des o.a. dritten Falls; q.e.d.

33) Es mag auf den ersten Blick befremden, daß hier nicht auf das gewöhnliche Begriffspaar "Ursache und Wirkung" Bezug genommen wird. Diese Abweichung folgt aber notwendig aus der dynamischen Eigenart des Petrinetz-Konzepts: Ein Netz wird dadurch verändert, daß seine aktuelle Markierung verändert wird. Dabei spielt es keine Rolle, ob sich die Markierungsveränderung auf die Anzahl, die Verteilung oder die Attributausprägungen von Markenkopien erstreckt. Die *Ursachen* der Markierungsveränderungen sind die Schaltakte von Transitionen. Die *Wirkungen* solcher Schaltakte werden durch die Schaltprozeduren derjenigen Transaktionen spezifiziert, die den geschalteten Transitionen jeweils eineindeutig zugeordnet sind. Die Transitionen können aber nur dann geschaltet werden, wenn ihre Aktivierungsbedingungen erfüllt sind. Daher besitzen die Aktivierungsbedingungen die Qualität von *Beeinflussungen.* Sie legen fest, ob die Schaltakte von Transitionen überhaupt zulässig sind. Daher beeinflussen die Aktivierungsbedingungen, ob Transitionen durch ihr Schalten - durch das Ausführen von Schaltprozeduren der transitionszugehörigen Transaktionen - Markierungsveränderungen verursachen und zugleich bewirken können.
Zuvor wurden Schaltakte von Transitionen als Ursachen der Veränderungen von Netzmarkierungen herausgestellt. Es liegt nahe, nun auch nach der Verursachung von Schaltakten zu fragen. An dieser Stelle zeichnet sich das Petrinetz-Konzept durch eine deutliche Zäsur aus: Der kausale Verursachungszusammenhang endet an den Schaltakten der Transitionen. Denn die Verursachung der Schaltakte wird innerhalb eines Netzes grundsätzlich nicht mehr erfaßt. Nur ihre Beeinflussung durch die Aktivierungsbedingungen der Transitionen findet Beachtung. Diese Eigenart resultiert aus der charakteristischen Permissivität der Schaltregel von Netzen. Sie wurde bereits an früherer Stelle erläutert. Dort wurde auch schon auf den Indeterminismus hingewiesen, den die Schaltregelpermissivität zur Folge hat. Permissivität und Indeterminiertheit der Schaltregel bedeuten, daß eine aktivierte Transition zwar schalten kann, aber nicht muß. Folglich besitzt die Aktivierungsbedingung einer Transition nur die Qualität einer notwendigen Voraussetzung für das Schalten der Transition. Sie erlangt aber nicht den Charakter einer hinreichenden Voraussetzung. Kausalursachen besitzen aber immer die Qualität von hinreichenden Voraussetzungen. Denn die Kausalwirkungen treten zwingend ein, wenn die Kausalursachen vorliegen. Daher ist die Aktivierungsbedingung einer Transition niemals deren Schaltursache. (Ein anderes Netzkonstrukt kommt erst recht nicht als Schaltursache in Betracht.) Allerdings umfaßt die Aktivierungsbedingung alle Aspekte der Verursachung des Schaltens einer Transition, die im Rahmen des Petrinetz-Konzepts netzendogen erfaßt werden können. Falls die Aktivierungsbedingung einer Transition erfüllt ist, braucht nur noch die netzexogene Information ergänzt werden, daß die Transition tatsächlich geschaltet werden soll. Netzendogene Aktivierungsbedingung und netzexogene Schaltinformation zusammen bilden die hinreichende Voraussetzung für das Erfolgen eines Schaltakts. Ihre Gesamtheit läßt sich als die Ursache des Schaltens einer Transition auffassen. Jedoch weist diese Schaltursache aufgrund ihrer netzexogenen Schaltinformation über das jeweils betrachtete Netz hinaus. Allerdings wird später eine Erweiterung Synthetischer Netze vorgelegt, die es erlaubt, diese Schaltinformation zu internalisieren. Es handelt sich um die obligatorische Schaltregel. Mit ihrer Hilfe läßt sich das Schalten von aktivierten Transitionen erzwingen. In solchen Fällen erlangen die Aktivierungsbedingungen der Transitionen den Charakter von hinreichenden Voraussetzungen. Sie können dann sogar als Ursachen der Transitionsschaltakte betrachtet werden.

34) Dabei werden die Begriffe "Beeinflussung" und "Wirkung" im Sinne von partiellen Beeinflussung und Wirkungen verwendet. "Die" Beeinflussung oder Wirkung des Schaltens einer Transition ist stets die Gesamtheit aus allen ihren partiellen Beeinflussungen bzw. partiellen Wirkungen. Diese Gesamtheit wird durch die lokale Netzstruktur

determiniert. Die lokale Netzstruktur enthält einerseits die Schaltprozedur der transitionszugehörigen Transaktion. Andererseits umfaßt sie die Prädikatssymbole und die aktuellen Prädikatsextensionen aller Stellen, die mit der jeweils betrachteten Transition benachbart sind.

35) Vgl. dazu den Ausschluß von Integritätsbedingungen aus dem Beeinflussungsbegriff.

36) Diese Trichotomie der kausalen Basis des Petrinetz-Konzepts stellt den tieferen Grund für die Schwierigkeiten dar, die oben mit der Eingliederung von Informationskanten in das Flußkonzept für die Kanteninterpretation gewöhnlicher Petrinetze auftraten. Gerichte Kanten können aufgrund ihrer Bipolarität zunächst nur dichotome Konzepte problemlos repräsentieren. Der Markenfluß ist ein solches dichotomes Konzept. Dieser Markenfluß besitzt aber aus kausaler Perspektive eine vermischte Struktur. Denn Eingangskanten in der einen Flußrichtung vermengen Beeinflussungen und Wirkungen des Schaltens einer Transition. Nur die Ausgangskanten, die in die andere Flußrichtung weisen, besitzen auch eine eindeutige Kausalinterpretation: Sie repräsentieren reine Wirkungen. Bei einer kausalen Interpretation von Petrinetzen stellen also nicht die Ein- und Ausgangskanten die originären Kantenkategorien dar. Ursprünglich definiert sind vielmehr die Informations- und Ausgangskanten, denen reine Beeinflussungen bzw. reine Wirkungen entsprechen. Die Eingangskanten werden erst daraus als kombinierte Repräsentationen von Beeinflussungen und Wirkungen abgeleitet.
Daraus ergeben sich zwei wesentliche Konsequenzen: Erstens sind Informationskanten keine inferiore "Zugabe" zu Synthetischen Netzen, sondern ein Basiskonstrukt ihrer kausalen Interpretation. Zweitens können die Flußkanten der etablierten Netzinterpretation solange nicht in reiner, unvermengter Weise kausal interpretiert werden, wie sie nicht um flußfreie Informationskanten erweitert werden. Bei konventionellen Arbeiten zum Petrinetz-Konzept werden solche Informationskanten jedoch nicht berücksichtigt. Daher wäre die Behauptung, daß Flußkonzept der Kanteninterpretation von konventionellen Petrinetzen besitze kausalen Charakter, strenggenommen nicht richtig.

37) Um den Anschluß zur etablierten Petrinetz-Terminologie zu wahren, wird jedoch weiterhin die Bezeichnung "Flußrelation" neben den o.a. neuen Relationsnamen zugelassen. Dabei wird das Relationspräfix allerdings nicht mehr wörtlich, sondern nur noch anschlußstiftend aufgefaßt. Darüber hinaus wird an der symbolischen Notation "F" für diese Relation festgehalten.

38) Da sich Netze durch ihre *variable* Beschriftung mit Netzmarkierungen auszeichnen, erstreckt sich das Enthaltensein von Marken auf alle zulässigen - d.h. erreichbaren - Markierungen eines Netzes.

39) Dies folgt aus der bijektiven Zuordnung von Stellen und Prädikatssymbolen durch die Beschriftungsfunktion bsp.

40) Dies folgt aus der bijektiven Zuordnung von Transitionen und Transaktionen durch die Beschriftungsfunktion btt.

41) Die Markierungen Synthetischer Netze werden zunächst immer in der hier präzisierten Weise auf Faktenmengen bezogen. Sie besitzen daher die Qualität von metasprachlichen Feststellungen über die Gültigkeit von atomaren prädikatenlogischen Formeln. Dieser Ansatz wird jedoch später differenzierter behandelt. Dort wird zwischen operationalen und deklarativen Netzmodellen unterschieden. Die faktenbezogene Markierungsinterpretation läßt sich dann nur noch für operationale Netzmodelle aufrechterhalten. In deklarativen Netzmodellen wird dagegen die Gültigkeit von prädikatenlogischen Formeln auf andere Weise ausgedrückt. Darüber hinaus setzen deklarative Netzmodelle im allgemeinen die Nullmarkierung als Ausgangsmarkierung voraus. Unter ihr sind alle Stellen des Netzmodells zunächst unmarkiert.
Es würde erheblichen Aufwand bereiten, die allgemeine Definition Synthetischer Netze sowohl auf den operationalen als auch auf den deklarativen Fall zu beziehen. Daher wird bei der Entfaltung des Kernkonzepts Synthetischer Netze - sofern keine ausdrücklich abweichenden Festlegungen erfolgen - stets der Kontext operationaler Netzmodelle vorausgesetzt. Dies gilt fortan immer, auch wenn darauf oftmals nicht mehr explizit hingewiesen wird. Die Fokussierung auf operationale Netzmodelle kann hier noch nicht gerechtfertigt werden. Sie setzt eine ausführliche Auseinandersetzung mit der deklarativen und der operationalen Modellierungsvariante voraus. Daher wird auf die Erläuterungen verwiesen, die später die Bevorzugung operationaler Netzmodelle motivieren.

42) Die Vorkommnisse desselben Prädikatssymbols können sich sowohl auf identische als auch auf verschiedene Prädikatsargumente erstrecken.

43) Es könnte die Ansicht vertreten werden, daß in diesem Fall überhaupt kein Synthetisches Netz vorliegt, weil eine Definitionskonstituente Synthetischer Netze fehlt. Dann müßten aber alle Netze, für die keine Ausgangsmarkierung vorgesehen ist, als Netzklasse sui generis behandelt werden. Um den daraus resultierenden terminologischen Aufwand zu vermeiden, läßt es der Verf. zu, auch solche Netze zu den Synthetischen Netzen zu rechnen.

44) Die nachfolgende Definition des Zeilenvektors als ein Element aus dem kartesischen Produkt der Mengen aller Multimengen von konstanten atomaren Formeln für die stellenzugehörigen Prädikatssymbole ist formal präziser als die originäre Definition des Nachbereichs der Markierungsfunktion M_r durch die Familie derselben Formelmultimengen. Denn nur das kartesische Produkt leistet explizit die korrekte Zuordnung der stellenzugehörigen Prädikats-

symbole. Bei der Markierungsfunktion M_r muß dagegen die Zuordnung der stellenzugehörigen Prädikatssymbole implizit als korrekt unterstellt werden.

45) Die Netzinterpretation ist formal, weil rein formalsprachlich definierte Netzkonstrukte herangezogen werden, um die Komponenten der topologischen Netzstruktur zu interpretieren. Auf eine materielle Netzinterpretation durch eine denotationale Netzsemantik wird später eingegangen.

46) Ein Spezialfall erstreckt sich auf Stellen s_m, denen ein Prädikatssymbol $Prä_u$(bas_marke) mit der Basismarke als Sorte des einstelligen Arguments zugeordnet ist. Der Definitionsbereich für alle zulässigen Kopien der Basismarke degeneriert zur einelementigen Menge $\{\varnothing\}$. Daraus folgt für die Notation der leeren Faktenmenge:

$$\forall(X \in \{\varnothing\}): \text{fakt}_{u,r}(0,\text{prä}_u(X)) \quad :\Leftrightarrow \quad FAK_{u,r} = \{\}$$

Da die Variable X nur durch die genau eine Konstante "∅" belegt werden kann, läßt sich die o.a. definitorische Äquivalenz noch weiter vereinfachen zu:

$$\text{fakt}_{u,r}(0,\text{prä}_u(\varnothing)) \quad :\Leftrightarrow \quad FAK_{u,r} = \{\}$$

Auf diese letzte Notationsweise wird später zurückgegriffen, um Integritätsbedingungen mit der Hilfe von faktischen Transitionen zu rekonstruieren.

47) Darüber hinaus weicht die allquantifizierte faktische Formel von der früheren Vereinbarung ab, faktische Formeln nur für *vorhandene* Markenkopien zu verwenden. Diese Abweichung wird jedoch aufgrund der nachstehend angedeuteten Vorzüge in Kauf genommen.

48) Die gleiche Kurznotation prädikatenlogischer Formeln findet sich bei DORN (1989), S. 54f. (dort allerdings für den Spezialfall von Intervallformeln).

49) Vgl. MURATA,TA. (1988b), S. 483f.

50) Dabei ist die Attributmarke $att\_marke_{s(u)}$ die Zielsorte eines Operationssymbols "$Marke_{j(u)}$" mit $j(u) \in \{1,...,J_A\}$.

51) Wegen m_0=$marke_0()$ wird das Formelvorkommnis $prä_u(m_0)$ bei vollständiger Termexplizierung als $prä_u(marke_0())$ notiert. Dies führte bei Anlehnung an die o.a. Behandlung von Attributmarken zu der Kurznotation <>, weil die Kopien der Basismarke keine innere (Attribut-)Struktur besitzen. Diese Kurznotation <> wiche aber formal von der oben eingeführten Kurznotation <∅> für Formelvorkommnisse $prä_u(m_0)$ ab. Um Mißverständnisse zu vermeiden, wird in dieser Arbeit darauf verzichtet, bei vollständiger Termexplizierung Kopien der Basismarke und Kopien von Attributmarken analog zu notieren. Statt dessen werden Formelvorkommnisse $prä_u(m_0)$, deren Argumente sich auf eine Kopie $m_0=\varnothing$ der Basismarke erstrecken, grundsätzlich nur als $<m_0)$ oder <∅> notiert (sofern eine verkürzte Darstellungsform gewählt wird).

52) Aufgrund der Voreinstellungen, die in den nachfolgenden Erläuterungen vereinbart werden, liegt das wesentliche Individualisierungspotential in der Differenzierung von Bestimmungsgleichungen und Haupttestbedingungen. Falls diese nicht spezifiziert bzw. durch die immer gültige Tautologie gebildet werden, erfolgt keine Differenzierung.

53) In welchem Ausmaß die Schaltregeln SR_v tatsächlich voneinander abweichen, hängt davon ab, wie stark die zugrundeliegenden Deklarationen der Transaktionen tr_v differieren. Vgl. dazu die Anmerkungen in der voranstehenden Erläuterung. Von der anderen Konstituente der Schaltregeln - dem allgemeinen Übergangsschema - kann per constructionem kein differenzierender Einfluß ausgehen.

54) Eine transitionsspezifische Schaltregel SR_n wird in dieser Arbeit auch als Schaltvorschrift der Transition t_n angesprochen.

55) In synonymer Weise kann die Deklaration einer Transaktion auch als die Transaktionsvorschrift ihrer zugehörigen Transition bezeichnet werden.

56) Mittelbar heißt hier, daß die Formeln for_z der Transition t_n nicht durch eine Beschriftungsfunktion direkt zugeordnet werden. Aber die Beschriftungsfunktion btt bildet die Transition t_n auf die Transaktion tr_v ab, die ihrerseits mit den Formeln for_z assoziiert ist.

57) Der Index "c" wird in Kürze als "Farbindex" (colour index) für unterschiedliche Schaltmodi von Transaktionen motiviert.

58) Dabei gilt es eine Besonderheit zu beachten, die sich auf Extremierungsformeln für die Auszeichnung minimaler oder maximaler Argumente erstreckt. Denn es kann nicht unmittelbar festgestellt werden, ob eine Variablenbelegungsfunktion vb_c eine Extremierungsformel for_z erfüllt. Statt dessen kann aufgrund des Extremierungscharakters der Formel for_z ihre Erfüllung durch eine Variablenbelegungsfunktion vb_c nur dadurch nachgewiesen werden, daß

diese Variablenbelegungsfunktion mit allen anderen zulässigen Variablenbelegungsfunktionen vb_k (mit k c) verglichen wird. Vgl. dazu die Erläuterung zur Globalität von extremierenden Optimierungsansätzen. Wie dieser Vergleich mit allen zulässigen Variablenbelegungsfunktionen realisiert wird, braucht hier nicht weiter untersucht zu werden. Es handelt sich um ein informationstechnisches Problem der Implementierung prädikatenlogischer Formeln.

59) Falls eine Präzisierung erwünscht ist, kann zwischen vollständig spezifizierten Schaltvoraussetzungen und -wirkungen einerseits sowie restriktionsformelabhängigen Schaltvoraussetzungen bzw. -wirkungen andererseits differenziert werden. Eine solche Unterscheidung bleibt aber so lange unnötig, wie aus dem jeweils aktuellen Argumentationskontext ersichtlich ist, zu welcher der beiden vorgenannten Kategorien die jeweils behandelten Schaltvoraussetzungen oder -wirkungen gehören.

60) Diese Kennzeichnung ist hilfreich, wenn die spezielle Darstellungsweise für Synthetische Netze verwendet wird, die im nächsten Kapitel vorgestellt wird. Denn dort werden in der Netzlegende die Restriktionsformeln lediglich aufgelistet. Alle Formeln, bei denen es sich um keine Gleichungen handelt, gehören notwendig zu einem Test. Restriktionsformeln in Gleichungsform können dagegen sowohl einen Test als auch eine Bestimmungsgleichung darstellen. Um diese Zweideutigkeit aufzulösen, wurde die o.a. Notationsweise festgelegt.

61) Dies folgt - allerdings in formal abweichender Darstellungsweise - aus einer Anwendung des "Distributionsgesetzes" bei GENRICH (1988b), S. 243.

62) Der Verf. bevorzugt in der später präsentierten Fallstudie diese erste Formulierungsalternative. Sie hat den Vorzug, daß die geforderte Attributausprägung durch eine entsprechende Haupttestbedingung kognitiv auffälliger notiert wird, als es bei der nachfolgend angesprochenen Integration in ein Kantengewicht der Fall wäre. Darüber hinaus entspricht die erste Formulierungsalternative der Notation von Prädikat/Transition-Netzen in Basisform, wie sie von GENRICH (1988b), S. 235, festgelegt ist (hinsichtlich des Fehlens symbolischer Koeffizienten in den Kantengewichten).

63) Es wird die vereinfachte Notation von Bestimmungsgleichungen verwendet. Die vollständige Notation lautet dagegen: $vb_c(At_{q.neu}) := vb_c(At_{q.alt})$.

64) Der Verf. präferiert diese zweite Formulierungsweise, wenn die Beibehaltung der Attributausprägung besonders hervorgehoben werden soll. Andernfalls zieht er die erste Ausdrucksvariante aufgrund ihrer Kompaktheit vor.

65) Die tautologische Formel gehört zu den impliziten Standardformeln aus der Formelmenge RES_v.

66) Einen Teilfall hiervon - die Festlegung der Bestimmungsgleichungen in der Gestalt "SIG-Gesetzen" - führt RECK (1988), S. 81 i.V.m. S. 79f., als allgemeine Beschriftung der Transitionen eines signaturbasierten Netzes an.

67) Dabei gehören die zweistelligen Vergleichsrelationen mit den Infixnotationen "=" und "≤" zu den Standardformel, die in der Formelmenge RES_v der Transaktion tr_v implizit enthalten sind.

68) Eine abweichende Kapazitätsdefinition für Netze auf algebraisch-prädikatenlogischer Grundlage findet sich bei REISIG (1989a), S. 37. Für dieselbe Netzkategorie sieht RECK (1988), S. 81, überhaupt keine Kapazitäten vor. Die Kapazitätsdefinition des Verf. stimmt dagegen mit der "Kapazität-B" überein, die im Programmpaket PASIPP für prädikatenlogisch basierte Netze definiert ist; vgl. OBERWEIS (1988a), S. 5 u. 7f.

69) Als solche Definitionsalternative kämen z.B. Mindestkapazitäten in Frage, auf die in der nachfolgenden Erläuterung zurückgekommen wird. Im Programmpaket PASIPP findet sich eine weitere Variante des Kapazitätskonzepts; vgl. OBERWEIS (1988a), S. 5 u. 7f. Dort wird eine "Kapazität-A" als die maximal zulässige Anzahl von *identischen* Kopien *desselben* (Marken-)Tupels definiert. Unterschiedliche (Marken-)Tupel können also durchaus einer Stelle in einer Anzahl zugeordnet werden, welche die "Kapazität-A" übersteigt. Die involvierten (Marken-)Tupel entsprechen in der Netzkonstruktion des Verf. jeweils einem Fakt.

70) Falls das Konkretisierungspotential des Übergangsschemas ÜS im Hinblick auf Markenkapazitäten und Posttests voll ausgeschöpft werden soll, kann die Beschriftungsfunktion bsk als eine Funktion mit zweistelligen Bildtupeln reformuliert werden. Dann bezeichnet die erste Tupelkomponente $KAP_{min.m}$ die Anzahl von Markentupeln, die auf der Stelle s_m mindestens erforderlich sind. Die zweite Tupelkomponente $KAP_{max.m}$ gibt dagegen - wie die ansonsten betrachtete Kapazität KAP_m - die höchstens zulässige Anzahl von Markentupeln auf der Stelle s_m an. Für die Mindestkapazität wird gefordert, daß sie niemals unendlich groß sein, wohl aber Null betragen darf. Dies entspricht dem intuitiven Verständnis von Mindestkapazitäten. Dann gilt für diese erweiterte Beschriftungsfunktion bsk und die hierauf fußenden Posttests:

$$\text{bsk:} \quad S \rightarrow \mathcal{N}_0 \times (\mathcal{N}_+ \cup \{\omega\})$$

$$s_m \rightarrow bsk(s_m) = (KAP_{min.m}, KAP_{max.m})$$

$$\text{posttest}(FAK_{u.+}) \;:\Leftrightarrow\; (bsk(s_m) = Prä_u \rightarrow KAP_{min.m} \leq \#(FAK_{u.+}) \leq KAP_{max.m})$$

71) Da die konventionelle Prädikatenlogik und Signaturkonzept in den komplementären Ansätzen einer sortierten Prädikatenlogik und einer prädikatenlogischen Signatur zusammengeführt wurden, braucht die Differenzierung zwischen prädikatenlogischen Variablenbelegungen und signaturbezogenen Variablenbindungsfunktionen nicht mehr aufrechterhalten zu werden. Daher werden Variablenbindungsfunktionen vb_c fortan synonym als Variablenbelegungen vb_c angesprochen.

72) Alle Ausgangsvariablen sind durch die Anwendung einer Variablenbindungsfunktion auf alle Einflußvariablen eindeutig determiniert. Dafür sorgt die Anschlußprämisse, die früher für das allgemeine Übergangsschema ÜS vorausgesetzt wurde. Sie gilt ebenso für die alle Transaktionen, die dieses Übergangsschema in einem Synthetischen Netz konkretisieren und den Transitionen zugeordnet werden.

73) Vgl. zu dieser farbbezogenen Redeweise, die sich für Höhere Netze weit verbreitet hat, z.B. PAGNONI (1990), S. 158 u. 160 (in bezug auf Prädikat/Transition-Netze). Ursprünglich stammt der Begriff der Schaltfarben aus dem Konzept der Gefärbten Netze. Mittlerweile ist er aber auch auf andere Höhere Netze übertragen worden.

74) Die Schaltfarbe "c_n" einer Transition t_n und die Schaltfarbe "c" einer Transaktion tr_v unterscheiden sich nicht wesentlich. Bei der Schaltfarbe "c" einer Transaktion tr_v wurde lediglich auf den Index "v" verzichtet, weil die Schaltfarbe "c" bereits als ein Index eingeführt wurde: als Index der Variablenbindungsfunktion vb_c. Die Indizierung eines Indexes unterblieb daher der Übersichtlichkeit halber. Anstatt von der Schaltfarbe c_n einer Transition t_n zu sprechen, könnte ebenso auf den Schaltmodus vb_c ihrer eineindeutig zugeordneten Transaktion tr_v Bezug genommen werden. Um die Unterscheidung zwischen Transaktionen und Transitionen zu unterstützen, wird aber vereinbart: Im Zusammenhang mit Transaktionen tr_v wird von Schaltmodi vb_c gesprochen. Im Kontext von Transitionen t_n werden dagegen Schaltfarben c_n verwendet.

75) Die maximalen Anzahlen zulässiger Schaltmodi und -farben könnten auch unendlich groß sein. Dies kommt immer dann in Betracht, wenn die Menge VBM aller Variablenbindungsfunktionen, die für ein Netz überhaupt definiert sind, unendlich ist. In der Netzspezifikation von Synthetischen Netzen wurde jedoch die Funktionenmenge VBM als eine endliche Menge vorausgesetzt. Denn für ihre Elemente - die Variablenbindungsfunktionen vb_c - wurde $c \in \{1,...,C\}$ mit $C \in \mathcal{N}_+$ festgelegt. Folglich können zwar beliebig viele, aber nur endlich viele verschiedene Variablenbindungsfunktionen in der Menge VBM enthalten sein. Diese Variablenbindungsfunktionen fallen mit den Schaltmodi einer Transaktion zusammen, sofern nur die Variablen dieser Transaktion betrachtet werden. Daher folgt aus der Endlichkeit der Menge VBM die Endlichkeit der maximalen Anzahl C_v zulässiger Schaltmodi für eine Transaktion tr_v. Da unter jeder erreichbaren Markierung die Anzahlen $C_{v.r}$ und $C_{n.r}$ zulässiger Schaltmodi bzw. -farben per constructionem identisch sind, muß auch die Anzahl C_n zulässiger Schaltfarben für eine Transition t_n endlich sein. Die hierbei vorausgesetzte Endlichkeit der Menge VBM wurde bereits an früherer Stelle für algebraisch-prädikatenlogische Spezifikationen begründet.

76) Analoges gilt für die Menge $MOD_{v,r}$ aller Schaltmodi vb_c einer Transaktion tr_v. Daher gelten alle nachfolgenden Erläuterungen, die sich explizit auf die Schaltfarben von Transitionen beziehen, implizit und mutatis mutandis ebenso für die Schaltmodi der zugehörigen Transaktionen.

77) Im selben Fall könnte ebenso von der Aktivierung der Transaktion tr_v unter der Markierung M_r hinsichtlich ihres Schaltmodus vb_c gesprochen werden: $AKT(tr_v,vb_c,M_r)$. Um die nachfolgenden terminologischen Erläuterungen nicht unnötig aufzublähen, wird auf solche transaktionsbezogenen Reformulierungsmöglichkeiten nicht mehr explizit eingegangen. Sie lassen sich aber bei Bedarf jederzeit nachholen, weil sie keine zusätzlichen Informationen erfordern.

78) Durch die Variablenbelegung vb_c, die zur Schaltfarbe c_n gehört, werden zunächst alle Variablen in denjenigen teilevaluierten Formeln gebunden, die in den Kantengewichten $bfm(s_m,t_n)$ enthalten sind. Dabei brauchen nur die Einflußkanten (s_m,t_n) der Transition t_n beachtet zu werden. Denn für das Schalten der Transition t_n müssen sich nur solche Markenkopien unter der Referenzmarkierung M_r auf der Stelle s_m befinden, die über Eingangskanten der Transition von dort abgezogen oder über Informationskanten der Transition dort zur Kenntnis genommen werden. Das Resultat $vb_c(bfm(s_m,t_n))$ ist eine Multimenge aus konstanten atomaren prädikatenlogischen Formeln. Diese Formeln werden durch die Anwendung des Faktoperators "fak_r" in faktische Formeln transformiert, die auf die Referenzmarkierung M_r bezogen sind. Falls der Stelle s_m durch die Beschriftungsfunktion bsp das Prädikatssymbol $Prä_u$ zugeordnet ist, nehmen die Formeln aus der Multimenge $fak_r(vb_c(bfm(s_m,t_n)))$ jeweils die Gestalt $fakt_r(mu_u,prä_u(ob_1,...,ob_{K_u}))$ an. Auch die aktuelle Markierung $M_r(s_m)$ der Stelle s_m ist eine Multimenge aus faktischen Formeln $fakt_r(mu_u,prä_u(ob_1,...,ob_{K_u}))$. Daher bereitet es keine Schwierigkeiten zu überprüfen, ab alle faktischen Formeln der Multimenge $fak_r(vb_c(bfm(s_m,t_n)))$ in der Multimenge $M_r(s_m)$ enthalten sind. Dies ist genau dann der Fall,

wenn die beiden Multimengen die Beziehung $fak_r(vb_c(bfm(s_m,t_n))) \leq M_r(s_m)$ [$\Leftrightarrow$ $M_r(s_m) \geq fak_r(vb_c(bfm(s_m,t_n)))$] erfüllen. Die Relatoren "$\leq$" [bzw. "$\geq$"] wurden für Multimengen bereits definiert.
Falls die Stelle s_m nicht zum Einflußbereich der Transition t_n gehört, gilt per definitionem $bfm(s_m,t_n)=\{\}$. Das leere Kantengewicht enthält keine Variable. Für die Variablenbindungsfunktion vb_c wird vereinbart, daß ihre Anwendung auf die leere Menge stets wieder die leere Menge hervorbringt: $vb_c(\{\})=\{\}$. Daher gilt für die Stelle s_m: $vb_c(bfm(s_m,t_n))=\{\}$. Analog dazu gilt die Festlegung, daß die Anwendung des Faktoperators auf die leere Menge immer die leere Faktenmenge erzeugt: $fak_r(\{\})=\{\}$. Also gilt für jede Stelle s_m, die nicht zum Einflußbereich der Transition t_n gehört: $fak_r(vb_c(bfm(s_m,t_n)))=fak_r(vb_c(\{\}))=fak_r(\{\})=\{\}$.

79) Bei der Überprüfung, ob durch das Schalten der Transition die Markenkapazität der Stelle s_m nicht überschritten wird, kommt es nur auf die Anzahlen der Markenkopien an. Daher wird hier die Zählfunktion "#" anstelle der Variablenbelegungsfunktion vb_c verwendet. Die Testbeschränkung auf Anzahlen von Markenkopien gestattet auch, auf den Faktoperator fak_r zu verzichten. Denn es kommt nur noch darauf an, *wie viele* Formelvorkommnisse durch Markenkopien auf der Stelle s_m repräsentiert werden. Ob es sich dabei um teilevaluierte oder konstante Formelvorkommnisse der objektsprachlichen Ebene oder um faktische Formelvorkommnisse der metasprachlichen Ebene handelt, spielt für die Feststellung der Formelvorkommnisse keine Rolle. Daher brauchen die Funktion vb_c und der Operator fak_r auf die Gewichte der Einfluß- und Ausgangskanten der Transition t_n nicht angewendet zu werden. Falls jedoch eine Angleichung der Formulierungen beiden Konjugatteile aus der Definition des Aktivierungsprädikats erwünscht ist, können die Kantengewichte $bfm(s_m,t_n)$ und $bfm(t_n,s_m)$ ebenso durch die komplexeren Konstrukte $fak_r(vb_c(bfm(s_m,t_n)))$ bzw. $fak_r(vb_c(bfm(t_n,s_m)))$ ersetzt werden. Dies würde die o.a. Definition des Aktivierungsprädikats lediglich durch folgende formal anders strukturierte, aber äquivalente Formulierung ersetzen:

$$AKT(t_n,c_n,M_r)$$

$$:\Leftrightarrow \quad \forall(s_m \in S): \quad (M_r(s_m) \geq fak_r(vb_c(bfm(s_m,t_n))))$$
$$\wedge \ (\#(M_r(s_m)) \leq KAP_m + \#(fak_r(vb_c(bfm(s_m,t_n)))) - \#(fak_r(vb_c(bfm(t_n,s_m)))))$$

Für beide äquivalenten Definitionen des Aktivierungsprädikats gilt: Falls die Stelle s_m nicht zum Einflußbereich (Nachbereich) der Transition t_n gehört, trifft per definitionem $bfm(s_m,t_n)=\{\}$ $(bfm(t_n,s_m)=\{\})$ zu. Dann werden die Kantengewichte - oder ihre Überformungen durch die Variablenbindungsfunktion vb_c und den Faktoperator fak_r - durch die Zählfunktion "#" jeweils auf den Wert Null abgebildet. Denn für die leere Multimenge "{}" gilt stets: $\#(\{\})=0$. Dies folgt unmittelbar aus der Definition der Zählfunktion.

80) Dieses Aktivierungsprädikat wird ebenso durch die Anweisungskomplexe definiert, die im Softwarepaket PASIPP die Auflistung aller (c_n,M_r)-aktivierten Transitionen erlauben. Dabei schlägt die imperative Semantik der Programmiersprache PROLOG eine Brücke zur prozeduralen Definition der Aktivierungsbedingung. Denn die PASIPP-Anweisungskomplexe stellen informationsverarbeitende Prozeduren dar, die sich genau dann ausführen lassen, wenn das deklarative Aktivierungsprädikat gültig ist.

81) Allerdings war das Aktivierungsprädikat $AKT(t_n,M_r)$ für Stelle/Transition-Netze originär definiert. Dagegen wird hier das Aktivierungsprädikat $AKT(t_n,M_r)$ für Synthetische Netze aus dem originär definierten Aktivierungsprädikat $AKT(t_n,c,M_r)$ und den Schaltfarben c_n abgeleitet.

82) Die Argumente, die an früherer Stelle hinsichtlich der Definitionsableitung vorgetragen wurden, gelten für Synthetische Netze unverändert. Sie werden daher hier nicht noch einmal wiederholt. Allerdings sind sie zu ergänzen im Hinblick auf den bereits oben dargelegten Sachverhalt, daß die Aktivierungsbedingungen von Transitionen nicht auf arithmetische Beziehungen zwischen Stellenmarkierungen, Kantengewichten und Markenkapazitäten reduziert werden können, sondern die Beachtung der gesamten Schaltprozeduren der jeweils zugeordneten Transaktionen erfordern. Dieser Umstand wird hier dadurch berücksichtigt, daß in Synthetischen Netzen das basale Aktivierungsprädikat $AKT(t_n,c_n,M_r)$ aus den abgeleiteten Prädikaten für die nebenläufige bzw. konfliktionäre Aktivierung nicht eliminiert werden kann. Bei Stelle/Transition-Netzen war dies dagegen möglich. Da die basalen Aktivierungsprädikate weiterhin benutzt werden müssen, entfallen jedoch nunmehr die beiden ersten Teilformeln aus der Definition des Prädikats konfliktionärer Aktivierung von Stelle/Transition-Netzen. Denn diese beiden Teilformeln stellten durch ihre Maximalitäts- und Minimalitätsanforderung lediglich sicher, daß die betrachteten Transitionen aktiviert waren. Da diese Transitionsaktivierung jetzt durch die Prädikate $AKT(t_n,c_n,M_r)$ gefordert wird, werden die beiden vorgenannten Teilformeln abundant. Daraus resultiert für Synthetische Netze eine Formulierung der nebenläufigen und konfliktionären Aktivierungsprädikate, die sich gegenüber der analogen Prädikatedefinition in Stelle/Transition-Netzen durch ein höheres - nunmehr vollkommenes - Ausmaß an Formulierungssymmetrie auszeichnet. Vgl. dazu die Definitionen der metasprachlichen Aktivierungsprädikate.

83) Auf die nebenläufige oder konfliktionäre Aktivierung von Transitionen unter Berücksichtigung von nicht-leeren Informationsbereichen wird in einer anschließenden Erläuterung näher eingegangen.

84) Für Informationskanten gilt die nachfolgende Überlegung nicht, weil die Marken von Informationsstellen durch die Transaktionen schaltender Transitionen nur zur Kenntnis genommen, aber nicht verändert werden. Dies spielt hier aber auch keine Rolle, weil zunächst Transitionen mit leeren Informationsbereichen vorausgesetzt wurden. Auch die Ausgangskanten bleiben unberührt. Auf ihren Ausgangsstellen werden zwar durch das Schalten der Transitionen Marken abgelegt. Aber für das Einhalten der Markenkapazitäten der Ausgangsstellen reichen die kapazitätsbezogenen Ungleichungen aus.

85) Falls sie keine gemeinsame Eingangsstelle besitzen, entfällt die hier vorgetragene Einschränkung.

86) Ein einfaches Beispiel mag diese Komplikation verdeutlichen. Sei s_m die Eingangsstelle zweier Transitionen t_x und t_y mit zugehörigem Prädikatssymbol $Prä_u(att\_marke_a)$. Unter der aktuellen Markierung M_r wird diese Stelle s_m von zwei Kopien $m_{s(j).d}=marke_j(at_{j.1.d},at_{j.2.d})$ mit $d \in \{1,2\}$ der zweistelligen Attributmarke $att\_marke_{s(j)}$ mit $s(j)=a$ belegt. Hierfür gilt:

$$m_{s(j).1} = fakt_r(1,prä_u(marke_j(at_{j.1.1},at_{j.2.1})))$$

$$m_{s(j).2} = fakt_r(1,prä_u(marke_j(at_{j.1.2},at_{j.2.2})))$$

Die Markierung der Eingangsstelle s_m nimmt dann als formale Summe folgende Gestalt an:

$$M_r(s_m) \;=\; fakt_r(1,prä_u(marke_j(at_{j.1.1},at_{j.2.1}))) + fakt_r(1,prä_u(marke_j(at_{j.1.2},at_{j.2.2})))$$

Die Eingangskanten (s_m,t_x) und (s_m,t_y) der beiden Transitionen t_x bzw. t_y sind mit Multimengen teilevaluierter Formeln beschriftet, die den Abzug jeweils einer Kopie der o.a. Attributmarke bedeuten. Diese Markenkopien werden fortan der Einfachheit halber nur als Marken angesprochen. Durch Verwendung der Variablen "At" für die Attributausprägungen "at" wird in den Kantenanschriften jedoch offengelassen, welche Markenkopie beim Schalten der Transitionen von der Eingangsstelle konkret abgezogen wird. Für diese Kantenanschriften gilt daher (wiederum in der Notation formaler Summen):

$$bfm(s_m,t_x) = bfm(s_m,t_y) = prä_u(marke_j(At_{j.1},At_{j.2}))$$

Die Markierung $M_r(s_m)$ der Eingangsstelle s_m reicht aus, um für beide Eingangskanten jeweils eine Marke für den schaltbedingten Markenabfluß bereitzustellen. Die rein anzahlbezogene Restriktion $\#(M_r(s_m)) \geq \#(bfm(s_m,t_x)) + \#(bfm(s_m,t_y))$ würde zwar durch die Markierung M_r wegen $\#(M_r(s_m))=2$ einerseits und $\#(bfm(s_m,t_x))=\#(bfm(s_m,t_y))=1$ andererseits erfüllt. Dennoch entsteht ein Zugriffskonflikt auf die Marken der gemeinsamen Eingangsstelle s_m, falls die Schaltfarben c_x und c_y der beiden Transitionen t_x bzw. t_y durch ihre Variablenbelegungen vb_{cx} bzw. vb_{cy} jeweils *dieselbe* Marke der Stelle s_m binden. Dies wäre z.B. der Fall für:

vb_{cx}:	$At_{j.1} \rightarrow at_{j.1.1}$		vb_{cy}:	$At_{j.1} \rightarrow at_{j.1.1}$
	$At_{j.2} \rightarrow at_{j.2.1}$			$At_{j.2} \rightarrow at_{j.2.1}$

Dann würde durch die Schaltfarben c_x und c_y versucht werden, *dieselbe* Marke $m_{s(j).1}=marke_j(at_{j.1.1},at_{j.2.1})$ von der Stelle s_m zweimal abzuziehen: einmal über die Eingangskante (s_m,t_x) zum Schalten der Transition t_x; das andere Mal über die Eingangskante (s_m,t_y) zum Schalten der Transition t_y. Dies ist aber tatsächlich nicht möglich, weil unter der Markierung M_r nur genau eine Marke $m_{s(j).1}=marke_j(at_{j.1.1},at_{j.2.1})$ die Stelle s_m belegt. Ein solcher Zugriffskonflikt wird erst dadurch ausgeschlossen, daß die Variablenbelegungen vb_{cx} und vb_{cy} für die beiden Transitionen t_x bzw. t_y zwei *verschiedene* Marken für den Markenabfluß festlegen. Eine solche konfliktfreie Kombination der beiden Variablenbelegungen liegt z.B. vor, wenn der Transition t_x durch ihre Variablenbelegung vb_{cx} die Marke $m_{s(j).2}$ und der Transition t_y durch ihre Variablenbelegung vb_{cy} die Marke $m_{s(j).1}$ zugeordnet wird:

vb_x:	$At_{j.1} \rightarrow at_{j.1.2}$		vb_y:	$At_{j.1} \rightarrow at_{j.1.1}$
	$At_{j.2} \rightarrow at_{j.2.2}$			$At_{j.2} \rightarrow at_{j.2.1}$

Den Ausschluß von Zugriffskonflikten auf abzuziehende Marken erfüllt die nachfolgend aufgestellte Definition der nebenläufigen Aktivierung zweier Transitionen dadurch, daß sie für deren Eingangsstellen die Variablenbelegungen vb_c explizit berücksichtigt.

87) Bei der formalen Definition konfliktionärer Aktivierung muß auf die Negate "$\neg(...\geq...)$" und "$\neg(...\leq...)$" der Vergleichsrelationen "$\geq$" bzw. "$\leq$" zurückgegriffen werden. Die naheliegende Verwendung der Gegenrelationen "$<$" bzw. "$>$" ist nicht zulässig, weil jene Gegenrelationen für Multimengen überhaupt nicht definiert wurden. Würde dies nachgeholt, so müßten die Gegenrelationen für *alle* verschiedenen Formelvorkommnisse gelten, die in den

Multimengen der Kantengewichte bfm(s_m,t_n) und bfm(t_n,s_m) vorkommen. Dies wäre aber eine *stärkere* Anforderung als die o.a. Negate der Relationen "≥" bzw. "≤". Daher würde der Gebrauch der Gegenrelationen zu einer fehlerhaften Definition konfliktionärer Aktivierung führen.

88) Bei der Definition der konfliktionären Aktivierung kann die erste Ungleichungsrelation "≥" aus der Definition der nebenläufigen Aktivierung nicht mit Hilfe einer Ungleichungsrelation "<" negiert werden. Denn die betroffenen Multimengen erfüllen - analog zu Vektoren - die Relationen "≥" und "<" nur dann, wenn in *beiden* Fällen die Multiplizitäten *aller* Elemente aus ihren Trägermengen diese Relationen erfüllen. Die Negation einer relationsbestimmenden Formel mit einem Allquantor für die involvierten Multiplizitäten liefert aber eine Formel mit einem Existenzquantor für die gleichen Multiplizitäten. Daher kann die Negation der Relation "≥" nicht in ein Negat der Formel "<" umgesetzt werden.

89) Damit werden Interferenzen von Eingangs- und Informationskanten, die von derselben Stelle aus zu unterschiedlichen Transitionen weisen, von vornherein ausgeschlossen. Solche Interferenzen würden drohen, falls es möglich wäre, daß eine Markenkopie beim Schalten mehrerer Transitionen, die zum selben Schaltschritt gehören, über die Eingangskante einer Transition abgezogen wird, während eine andere Transition versucht, dieselbe Markenkopie über eine Informationskante zur Kenntnis zu nehmen. In diesem Fall wäre für die zweite Transition der Zugriff auf die Markenkopie, die von der ersten Transition abgezogen wird, nicht mehr wohldefiniert. Dieses Zugriffsproblem wird aber dadurch verhindert, daß Transitionen nur dann nebenläufig aktiviert sind, wenn gilt: Auf jeder ihrer gemeinsamen Eingangsstellen stehen genügend Markenkopien zur Verfügung, um sowohl den Markenabzug über die Eingangskanten der Transitionen als auch die Markenkenntnisnahme über die Informationskanten der Transitionen abzudecken. Genau dies leistet die nachfolgende verallgemeinerte Definition der nebenläufigen Aktivierung mehrerer Transitionen. Vgl. auch in der später präsentierten Fallstudie die Netzgraphik, die zum Netzmodul für die dritte Kernnetzerweiterung bei der Modellierung einer Bearbeitungsstation gehört. Dort greifen die Transition t_{14} und die Transition t_{17} auf dieselbe Kopie einer Puffermarke zu, die sich auf der Stelle s_{21} befindet. Die zweite Transition ist mit dieser Stelle über eine Informations-, die erste (u.a.) über eine Eingangskante verbunden. Daher würden die oben geschilderten Interferenzen drohen, falls beide Transitionen unter derselben Netzmarkierung aktiviert sind. Allerdings wird in der Fallstudie die gleichzeitige Aktivierung der beiden Transitionen durch die eine Kopie der Basismarke, die über die Stellen s_{19} und s_{23} fließt, von vornherein verhindert. Daher vermag sich das hier geschilderte Interferenzproblem dort nicht auszuwirken.

90) Dies läßt sich anhand eines einfachen Beispiels verdeutlichen: Drei Transitionen t_1, t_2 und t_3 besitzen eine gemeinsame Eingangsstelle, von der sie über ihre Eingangskanten jeweils eine Marke abziehen würden, ohne auf der Stelle eine Marke wieder abzulegen. Auf der Eingangsstelle befinden sich zwei Marken derart, daß je zwei der vorgenannten Transitionen nebenläufig aktiviert sind (Dabei wird der Übersichtlichkeit halber von anderen Stellen aus den Nachbarschaften der drei Transitionen abgesehen, die sich ebenso auf deren Aktivierungen auswirken können.) Dann wäre eine nebenläufige Aktivierung aller drei Transitionen ausgeschlossen, weil hierzu auf ihrer gemeinsamen Eingangsstelle mindestens drei Marken erforderlich wären.

91) Diese Festlegung ist zwar nicht notwendig. Doch erleichtert sie die formale Handhabung von Schaltschritten und deren materielle Interpretation. Theoretisch läßt sich durchaus vorstellen, daß eine Transition im selben Schaltschritt mehrfach enthalten ist. Dies kann immer dann der Fall sein, wenn eine Transition unter derselben Markierung durch mehrere Schaltfarben multipel aktiviert ist. Eine Transition ist nebenläufig zu sich selbst aktiviert, falls zwei Bedingungen erfüllt sind: Erstens muß sie bezüglich mehrerer Schaltfarben aktiviert sein. Zweitens müssen die Marken auf ihren Eingangsstellen und die freien Markenkapazitäten auf ihren Ausgangsstellen ausreichen, um die Transition mit mindestens zwei aktivierenden Schaltfarben zugleich oder in jeder beliebigen Reihenfolge zu schalten. Um diesen Sonderfall formal zu erfassen, wäre es allerdings erforderlich, die unten angeführten Definitionen für die nebenläufige und konfliktionäre Aktivierung von Transitionen erheblich komplizierter zu formulieren. Darüber hinaus müßten zusätzliche Definitionen für die in sich konfliktionäre bzw. nebenläufige Aktivierung einer einzelnen, aber multipel aktivierten Transition ergänzt werden. Schließlich wäre das zu sich selbst nebenläufige Schalten einer multipel aktivierten Transition nur schwer inhaltlich zu verstehen. Denn das Schalten einer Transition stellt im Petrinetz-Konzept ein atomares, nicht weiter zerlegbares Ereignis dar. Falls dieselbe Transition nebenläufig zu sich selbst schalten dürfte, müßte geklärt werden, ob dann noch ein atomares Schaltereignis stattfindet oder ob das Schalten derselben Transition zu einer Gruppe von separaten Schaltakten zerfließt. Um derartige formale und materielle Komplikationen zu vermeiden, wird in dieser Arbeit ausgeschlossen, daß eine Transition nebenläufig zu sich selbst aktiviert ist. Dann kann eine Transition auch niemals nebenläufig zu sich selbst schalten und somit auch niemals mehrfach im selben Schaltschritt enthalten sein. Dem Verf. ist auch kein Beispiel aus der Literatur zu Höheren Netzen bekannt, in dem das nebenläufige Schalten einer multipel aktivierten Transition mit sich selbst erörtert - geschweige denn zugelassen - würde.
Der Ausschluß multipler Aktivierungen derselben Transition konstituiert einen neuartigen Abundanzkonflikt. Abundanzkonflikte wurden bereits für Stelle/Transition-Netze eingeführt. Dort ließen sie sich aber nur auf nichtmaximale Schaltschritte beziehen, deren Vereinigung wiederum einen Schaltschritt darstellt. Die betroffenen Schaltschritte mußten daher disjunkte Transitionenmengen darstellen. Hier wird dagegen ein Abundanzkonflikt betrachtet,

der sich auf nur eine Transition bezieht. Er betrifft eine Transition, die bezüglich mehrerer Schaltfarben aktiviert ist. Falls diese Transition tatsächlich geschaltet werden soll, besteht ein Auswahlproblem. Denn es muß entschieden werden, mit welcher der aktivierenden Schaltfarben das Schalten der Transition erfolgen soll. Dieser Konflikt besitzt den Charakter einer Abundanz, weil mehr aktivierende Schaltfarben existieren, als zum Schalten der Transition benötigt werden. Abundanzkonflikte verhalten sich komplementär zu den früher definierten Knappheitskonflikten. Knappheitskonflikte treten immer dann auf, wenn *mehrere* aktivierte Transitionen auf dieselben Marken oder dieselbe freie Markenkapazität so zugreifen, daß es *unzulässig* ist, alle Transitionen zugleich mit jeweils *einer* Schaltfarbe zu schalten. Abundanzkonflikte stellen sich dagegen ein, sobald für *eine* Transition so viele Marken und freie Kapazität zur Verfügung stehen, daß es *zulässig* wäre, die Transition zugleich mit *mehrfachen* Schaltfarben zu schalten.

92) Es wurde im Rahmen der prädikatenlogischen Fundierung vereinbart, daß die Prädikatsnamen jedes Prädikatssymbol - unabhängig von der Stelligkeit der Prädikatsargumente - eindeutig identifizieren sollen. Daher müßte hier das schaltschrittbezogene Aktivierungsprädikate vom bereits eingeführten transitionenbezogenen Aktivierungsprädikat durch einen differierenden Prädikatsnamen unterschieden werden. Hiervon wird jedoch aus zwei Gründen abgesehen. Erstens wird im Rahmen der Prädikatenlogik auch zugelassen, namensgleiche Prädikatssymbole durch verschiedene Stelligkeiten voneinander zu unterscheiden. Dies würde für die Identifizierung der schaltschritt- und transitionenbezogenen Aktivierungsprädikate ausreichen, da erstes zwei- und letztes dreistellig definiert ist. Folglich brauchte nur die o.a. Vereinbarung widerrufen zu werden. Zweitens lassen sich die Vereinbarungen des prädikatenlogischen Fundaments auf die prädikatenlogische Repräsentation des Wissens über ein zu modellierendes Objekt beschränken. Die hier thematisierten metasprachlichen Aktivierungsprädikate gehören nicht zu solchen objektsprachlichen Prädikaten. Da die metasprachlichen Prädikate durch das Softwarepaket PASIPP implementiert werden, sind sie von der separaten prädikatenlogischen Objektrepräsentation wohlunterschieden. Diesen zweiten Ausweg bevorzugt der Verf. Allerdings müßte das Softwarepaket PASIPP noch um metasprachliche Konstrukte für schaltschrittbezogene Aktivierungsprädikate erweitert werden, da es bislang nur die Aktivierung und das Schalten jeweils einer Transition berücksichtigt.

93) In der Notation des Ausführens eines Schaltschritts SS_a tauchen die Schaltfarben der schaltschrittzugehörigen Transitionen nicht mehr explizit auf, weil sie implizit in den Schaltpaaren $(t_{n(w)},c_{n(w)})$ des Schaltschritts enthalten sind.

94) Dies wurde bereits für Stelle/Transition-Netze ausführlicher erläutert. Die dort vorgetragenen Argumente hinsichtlich der Invarianz des Ausführungsergebnisses bei alternativen Schaltfolgen gelten für Synthetische Netze unverändert. Allerdings kann die Schaltwirkung des Schaltschritts SS_a nicht mehr additiv aus den Schaltwirkungen der einzelnen Transitionen zusammengesetzt werden. Denn die Schaltwirkung läßt sich nicht mehr auf rein arithmetisch definierte Stellenmarkierungen, Kantengewichte und Markenkapazitäten reduzieren. Darauf wurde schon mehrfach hingewiesen. Statt dessen muß nun die Schaltprozedur der transitionszugehörigen Transaktion ausgeführt werden, um die Schaltwirkung jeder schaltschrittzugehörigen Transition zu ermitteln.
Die Schaltprozeduren können bei einem nicht-degenerierten Schaltschritt allerdings nebenläufig ausgeführt werden. Denn die Aktivierung eines nicht-degenerierten Schaltschritts wurde oben so eingeführt, daß seine Transitionen nebenläufig aktiviert sein müssen und hierbei die Multimengen aller transitionszugehörigen Eingangskanten unter den jeweils ausgewählten Schaltfarben in den Markenmultimengen der zugehörigen Eingangsstellen enthalten sind. Dieses nebenläufige Ausführungspotential läßt sich jedoch erst bei der Implementierung von Synthetischen Netzen in nebenläufigen Informationsverarbeitungssystemen berücksichtigen. Der hier gewählte arithmetische Kalkül für die Definition der Schaltwirkung besitzt nicht die Aussagekraft, die Nebenläufigkeit von Ermittlungsoperationen auszudrücken. Daher muß hier als Nebenläufigkeitssubstitut die Formulierung gewählt werden, jede beliebige Schaltfolge der schaltschrittzugehörigen Transitionen zuzulassen. Auch das Softwarepaket PASIPP verfügt noch nicht über die Fähigkeit, nebenläufige Netzkonstrukte zu implementieren. Allerdings wurde bereits auf Ansätze hingewiesen, die Programmiersprache PROLOG, die auch PASIPP zugrundeliegt, um nebenläufige Prozedurausführungsfähigkeiten zu bereichern. Darüber hinaus hat der Verf. in seiner algorithmischen Beschreibung von Erreichbarkeitsanalysen ausführlich von diesem Nebenläufigkeitspotential des Petrinetz-Konzepts Gebrauch gemacht.

95) Für jedes Schaltpaar $(t_{n(w)},c_{n(w)})$ wird genau eine Schaltprozedur ausgeführt. Es handelt sich um die Schaltprozedur der transitionszugehörigen Transaktion tr_v mit $btt(t_{n(w)})=tr_v$. Die Unterbestimmtheit der Prozedurausführung wird durch die Schaltfarbe $c_{n(w)}$ beseitigt, da sie genau eine Variablenbelegung vb_c für die Schaltprozedur bestimmt. Jede Schaltpaarauflistung bedeutet eine andere Reihenfolge für die sequentielle Abarbeitung aller Schaltprozeduren. Daher bedeutet eine Permutation der Schaltpaarauflistung zugleich, daß die Schaltfolge der schaltschrittzugehörigen Transitionen variiert.

96) Ebenso läßt sich das Paar (tr_v,vb_c) betrachten, das mit dem degenerierten Schaltschritt $SS_a=\{(t_n,c_n)\}$ korrespondiert, falls die Transaktion tr_v der Transition t_n zugeordnet ist. Aus dieser Perspektive kann der Schaltakt $sa_{r.a.f}$ auch als $sa_{r.v.c.f}$ notiert werden. Dann werden im Index der Schaltaktnotation die Transaktion tr_v und ihre Variablenbelegung vb_c unmittelbar ausgewiesen. Dabei entspricht die Variablenbelegung vb_c stets der o.a. Schaltfarbe c_n.

97) Modal- und Aktionslogik wurden bereits an früherer Stelle angesprochen. Der modallogische Aspekt von Schaltakten bezieht sich auf deren Potentialität: Ein Schaltschritt, der unter der aktuellen Markierung aktiviert ist, aber (noch) nicht ausgeführt wurde, unterliegt der modalen Kategorie der Möglichkeit: Der Schaltschritt *kann* ausgeführt werden. Die Verwandlung eines potentiellen Schaltakts in einen realisierten Schaltakt beendigt dagegen den Schwebezustand der modalen Möglichkeit. Zugleich bedeutet sie das tatsächliche Ausführen des zugrundeliegenden Schaltschritts. Dadurch wird die Aktion des Schaltens aller Transitionen aus dem Schaltschritt verwirklicht. Dies überführt nicht nur modallogische Potentialität in Realität. Vielmehr wird auch das Kernkonstrukt von Aktionslogiken, das Ausführen von Aktionen (Handlungen) direkt erfaßt. Daher wurde auch oben zwischen dem Schaltschritten und Schaltakten deutlich unterschieden: Ein Schaltschritt kann aktiviert sein (muß es aber nicht); falls er aktiviert ist, braucht er nicht ausgeführt zu werden. Einem Schaltschritt kommt also vornehmlich das modallogische Moment der Möglichkeit zu. Erst wenn der aktivierte Schaltschritt tatsächlich ausgeführt wird, verdichtet er sich zu einem punktförmig-ereignishaften Schaltakt. In diesem Punkt vollzieht sich der Übergang von der Möglichkeit zur Wirklichkeit durch das Geschehnis einer Aktion.
Es ist nicht Anliegen dieser Arbeit, die zuvor skizzierten Ansätze zur Erweiterung um modal- und aktionslogische Aspekte zu vertiefen. Sie verdeutlichen aber das überaus reichhaltige formallogische Ausdruckspotential des Petrinetz-Konzepts. Es wird zwar von der Prädikatenlogik 1. Stufe dominiert, eröffnet aber auch die Möglichkeit, dieses konventionelle Fundament subtiler fortzuentwickeln. Vgl. darüber hinaus auch die Anmerkungen zur modallogischen Qualität der Erreichbarkeitsgraphen, welche die dynamische Struktur eines Netzes repräsentieren.

98) Allerdings wird auf die explizite Angabe der Schaltregel SR_S in der Notation $RM(M_0,SR_S)$ für Erreichbarkeitsmengen verzichtet. Denn im Gegensatz zu Stelle/Transition-Netzen stützen sich Synthetische Netze nicht mehr auf eine universelle Schaltregel SR_S. An ihre Stelle treten das allgemeine Übergangsschema ÜS und dessen Konkretisierungen durch die Spezifikationen der Transaktionen tr_v. In ihrer Gesamtheit bedeuten Übergangsschema und Transaktionsspezifikationen individualisierte Schaltregeln für jede einzelne Transition eines Synthetischen Netzes. Die Gesamtheit aller dieser transitionsspezifischen Schaltregeln wird in der vereinfachten Notation $RM(M_0)$ für Erreichbarkeitsmengen von Synthetischen Netzen fortan als implizit bekannt vorausgesetzt.

99) Gleiches gilt für die Kontrollstruktur eines Synthetischen Netzes. Es wurde bereits aufgezeigt, daß die Kontrollstruktur vollständig festliegt, wenn der Erreichbarkeitsgraph eines Netzes gegeben ist.

100) Beispielsweise ist die Menge $pot_+(T)$ aller denkmöglichen Schaltschritte aus Stelle/Transition-Netzen in Synthetischen Netzen durch die Menge SSM zu ersetzen. Darüber hinaus läßt sich die Schaltregel von Synthetischen Netzen aufgrund der komplexen prozeduralen Charakteristik des Übergangsschemas ÜS nicht mehr als eine einfache Schaltregelfunktion SR_s wie bei Stelle/Transition-Netzen ausdrücken. Daher wird die Wirkung von Schaltschrittausführungen durch die oben vorgestellten Notationen für Schaltschritte und -folgen in Synthetischen Netzen reformuliert.

101) Dies wurde schon ausführlicher für die Erreichbarkeitsgraphen von Stelle/Transition-Netzen erläutert.

102) Es werden zwei Transitionen betrachtet, welche dieselben Einflußstellen, dieselben Ausgangsstellen und dieselben Kantenanschriften besitzen. Dann sind diese beiden Transitionen notwendig unter denselben Markierungen M_r mit denselben Schaltfarben aktiviert. Dabei wird vorausgesetzt, daß die beiden Transitionen dieselben Prä-, Haupt- und Posttests besitzen. Diese Einschränkung ist zulässig, weil die oben entfaltete Argumentation nur die Existenzmöglichkeit mindestens einer Multikante im Erreichbarkeitsgraphen für mindestens ein Synthetisches Netz nachzuweisen versucht. Mindestens ein Synthetisches Netz erfüllt aber die vereinfachenden Voraussetzungen für die Prä-, Haupt- und Posttests.
Dennoch können sich die beiden Transitionen in ihren Schaltwirkungen dadurch unterscheiden, daß ihre zugehörigen Transaktionen Restriktionsformeln mit unterschiedlichen Bestimmungsfunktionen für ihre Ausgangsvariablen besitzen. Diese unterschiedlichen Bestimmungsfunktionen werden im Regelfall dazu führen, daß die Schaltprozeduren der transitionszugehörigen Transaktionen trotz gleicher Variablenbelegungen ihrer Einflußvariablen die Ausgangsvariablen mit unterschiedlichen Werten binden. Daher werden sich die Folgemarkierungen der beiden Transitionen im allgemeinen auch dann unterscheiden, wenn die Transitionen unter denselben Markierungen M_r aktiviert sind und mit denselben Variablenbelegungen für ihre Einflußvariablen geschaltet werden. Daher wird das zweite Bijugat des Extensionalitätsaxioms in der Regel verletzt. Folglich sind die beiden Transitionen tatsächlich verschieden. Hierin liegt der wesentliche Unterschied zu Stelle/Transition-Netzen.
Dennoch kann bei solchen nicht-identischen Transitionen folgender Sonderfall eintreten: Ihre unterschiedlichen Bestimmungsfunktionen binden die Ausgangsvariablen der Transitionen unter einer speziellen Markierung M_r bezüglich einer speziellen Belegung "c" der Einflußvariablen so, daß gilt: Jede Ausgangsvariable der einen Transition wird mit demselben Wert gebunden wie die entsprechende Ausgangsvariable der anderen Transition. Dann führen beide Transitionen, die unter derselben Markierung M_r aktiviert waren, durch ihr Schalten mit derselben Belegung "c" ihrer Einflußvariablen dieselbe Folgemarkierung M_f herbei, obwohl ihre zugehörigen Transaktionen unterschiedliche Bestimmungsfunktionen besitzen. In diesem Sonderfall existiert im Erreichbarkeitsgraphen des Synthetischen Netzes eine Multikante: Die eine zugehörige Kante $ka_{r.f.1} = (M_r,M_f)$ ist mit dem Schaltakt (t_1,c) der ersten

Transition, die zweite Kante $ka_{r.f.2}=(M_r,M_f)$ mit dem Schaltakt (t_2,c) der zweiten Transition beschriftet. Der resultierende Erreichbarkeitsgraph ist ein Multigraph; q.e.d.

103) Falls Für Synthetische Netze nur das Schalten einzelner Transitionen zugelassen wird, läßt sich ein analoges Extensionalitätsaxiom wie für Stelle/Transition-Netze aufstellen. Es ist lediglich um den Aspekt der Variablenbelegungen "c" erweitert. Ihm zufolge gelten zwei Transitionen als identisch, falls sie unter denselben Referenzmarkierungen M_r mit denselben Schaltfarben "c" aktiviert sind und hierbei jeweils dieselben Folgemarkierungen M_f hervorbringen:

$$\forall(t_1\in T)\ \forall(t_2\in T): \ldots$$

$$(\forall M_r\in RM(M_0)\ \forall M_f\in RM(M_0): (MOD_{1.r}=MOD_{2.r}=MOD_r \wedge \ldots$$

$$(\forall(c\in MOD_r): (AKT(t_1,c,M_r) \leftrightarrow AKT(t_2,c,M_r)) \wedge (M_r\,[t_1,c\rangle\, M_f \leftrightarrow M_r\,[t_2,c\rangle\, M_f)))$$

$$\rightarrow\quad t_1=t_2$$

104) Sie wird auch kurz als Netzstatik bezeichnet.

105) Vgl. dazu die anlogen Ausführungen zur Strukturdifferenzierung von Stelle/Transition-Netzen.

106) Sie wird auch kurz als Netzdynamik bezeichnet.
Die Netzdynamik besitzt aus prädikatenlogischer Sicht eine besondere Qualität: Sie läßt sich als eine netzspezifische "dynamische" Logik interpretieren. Ausgangspunkt dieser Sichtweise ist die Faktenmenge FAK_0 der Ausgangsmarkierung M_0. Sie umfaßt alle atomaren Formelvorkommnisse, die unter der Ausgangsmarkierung des Netzes gültig sind. Der Verbund aus dem allgemeinen Übergangsschema ÜS und seiner Konkretisierung durch die transitionszugehörigen Transaktionen tr_v legt die Gesamtheit aller Faktenmengen FAK_r fest, die aus der Faktenmenge FAK_0 der Ausgangsmarkierung durch das Schalten von Transitionen hervorgebracht werden können. Auf diese Weise wird die logische Operation der Inferenz neuer gültiger Formelvorkommnisse aus alten gültigen Formelvorkommnissen auf die netzbezogene Operation der Konstruktion zulässiger Schaltprozesse zurückgeführt. Die Schaltregel eines Netzes erlaubt daher eine dynamische Operationalisierung des logischen Inferenzbegriffs. Später wird gezeigt, wie sich Netze so ausgestalten lassen, daß sie die volle Leistungskraft des prädikatenlogischen Inferenzkonzepts der kombinierten Unifizierung und Resolution erhalten. Vgl. dazu die Ausführungen zu deklarativen Netzmodellen.

107) Die nachfolgend definierten Netzzustände und -verhaltensweisen lassen sich ebenso als Modellzustände bzw. -verhaltensweisen behandeln, falls ein Synthetisches Netz in der Funktion eines Netzmodell verwendet wird.

108) Synonym wird auch von kombinatorisch möglichen oder denkmöglichen Netzzuständen (Modellzuständen) gesprochen.

109) Synonym läßt sich ebenso von kombinatorisch möglichen oder denkmöglichen Netzverhaltensweisen (Modellverhaltensweisen) reden.

110) Falls zwischen zwei unmittelbar aufeinander folgenden Markierungen M_r und M_f der Schaltschritt SS_a vermittelt, so wird nicht unterstellt, daß der Schaltschritt SS_a unter der Referenzmarkierung M_r aktiviert sein muß. Ebensowenig wird vorausgesetzt, daß die Folgemarkierung M_f die korrekte Schaltwirkung des Schaltschritts SS_a darstellt, wenn er auf die Referenzmarkierung M_r angewendet wird. Je nachdem, ob die Aktivierungsvoraussetzung und die Wirkungskorrektheit immer erfüllt werden oder mindestens einmal verletzt sind, wird nachfolgend zwischen zulässigen bzw. unzulässigen Netzverhaltensweisen unterschieden.

111) Dabei *konstituieren* die explizierten Integritätsbedingungen die Zulässigkeitsanforderungen in formaler Weise. Unter welchen Bedingungen ein Synthetisches Netz zulässig ist, ist nicht schon an anderer Stelle formal definiert worden, sondern wird mit der Erfüllung aller Integritätsbedingungen aus der Menge IB originär festgelegt.

112) Bei MURATA,TA. (1988b), S. 484, Definition 3.7, Unterfall 3), findet sich eine weitere Integritätsbedingung für Prädikat/Transition-Netze. Sie drückt aus, daß alle Ein- und Ausgangskanten derselben Stelle nur mit K-Tupeln *gleicher* Stelligkeit "K" beschriftet werden dürfen. Diese K-Tupel entsprechen bei Synthetischen Netzen den Argumenten derjenigen prädikatenlogischen Formeln $prä_u(te_1,...,te_{K_u})$, die zu den kantenbeschriftenden Multimengen $MTAV_{u.n}$, $MTAI_{u.n}$ und $MTAN_{u.n}$ gehören. Die Stelligkeit der Argumente dieser Formeln ist in Synthetischen Netzen bereits durch due K_u-Stelligkeit desjenigen Prädikatssymbols $Prä_u$ eindeutig definiert, das einer Stelle s_m durch die Beschriftung $bsp(s_m)=Prä_u$ zugeordnet ist. Alle adjazenten Kanten dieser Stelle nehmen in ihren o.a. Gewichten $MTAV_{u.n}$, $MTAI_{u.n}$ und $MTAN_{u.n}$ auf dieses Prädikatssymbol $Prä_u$ Bezug. Daher braucht die K_u-Stelligkeit der Formeln aus diesen Kantengewichten in Synthetischen Netzen nicht mehr durch eine separate Integritätsbedingung sichergestellt zu werden.

113) Eine Netztopologie ist genau dann konkret bestimmt, wenn ihre Stellen-, Transitionen- und Kantenmenge explizit vorliegen.

114) Falls das Prädikatssymbol $Prä_u$ für die Stelle s_m sowohl zum Vor- als auch zum Informationsbereich der Transaktion tr_v für die Transition t_n gehören würde, müßte dieselbe Kante (s_m,t_n) mit zwei verschiedenen Multimengen beschriftet werden. Dann wäre die Zuordnung der Multimengen aber eine rechtsmehrdeutige Beschriftungsabbildung.

115) Auf die Verknüpftheitsbedingung für Synthetische Netze wurde in diesem Kapitel eingegangen. Die entsprechende Verknüpftheitsbedingung wurde für Petrinetze (i.e.S.) schon vorgestellt.

116) Solche wirkungsfreien Transitionen lägen in Synthetischen Netzen immer dann vor, wenn sie zwei Bedingungen erfüllen: Erstens müssen sie mindestens eine benachbarte Stelle besitzen. Dadurch erfüllen sie die konstitutive Verknüpftheitsbedingung. Zweitens dürfen die Transitionen mit allen ihren inzidenten Stellen nur über Informationskanten verknüpft sein. Informationskanten lösen in Synthetischen Netzen die 1-Schleifen aus gewöhnlichen Netzen ab. Daher fallen die hier erwogenen Transitionen mit den oben diskutierten Transitionen aus Petrinetzen (i.e.S.) zusammen, die mit ihren inzidenten Stellen ausschließlich 1-Schleifen bilden. In Synthetischen Netzen werden die vorgenannten Transitionen aber kraft der Wirkungsbedingung ausgeschlossen. Denn Transitionen, die mit ihren inzidenten Stellen nur über Informationskanten verknüpft sind, besitzen notwendig zugeordnete Transaktionen mit leeren Wirkungsbereichen. Genau dies wurde aber verboten.

117) Die Bedeutung von Integritätsbedingungen für die Modellierung von Prozeßkoordinierungen in Produktionssystemen wurde bereits anläßlich der systemtheoretischen Strukturierung solcher Koordinierungsaufgaben angesprochen. Im Rahmen des Petrinetz-Konzepts werden diese Integritätsbedingungen für Modellierungsobjekte auf die hier vorgestellten netzspezifischen Integritätsbedingungen abgebildet. Diese Abbildung ist insofern bemerkenswert, als die formale Definition der Integritätsbedingungen aus der Menge IB grundsätzlich einen globalen, auf ein Netz insgesamt bezogenen Charakter besitzt. Dagegen zeichnet sich das Petrinetz-Konzept sonst durch seine rein lokal definierten Konstrukte - insbesondere durch die nur lokal wirksame Schaltregel von Transitionen - aus. Diese Lokalität des Petrinetz-Konzepts wird an anderer Stelle ausführlicher behandelt. Der scheinbare Widerspruch zwischen lokalem Modellierungsansatz des Petrinetz-Konzepts im allgemeinen und globaler Modellierungscharakteristik von netzspezifischen Integritätsbedingungen andererseits läßt sich jedoch später auflösen. Dort kann aufgezeigt werden, daß der Ausdrucksreichtum des Petrinetz-Konzepts ausreicht, um auch die global definierten netzspezifischen Integritätsbedingungen in seine primär lokale Modellierungsweise einzubeziehen.

118) Vgl. den Hinweis zum schematischen Charakter der hier vorgelegten Netzdefinitionen.

119) Darüber hinaus läßt sich die Sektion "trans" dadurch vereinfachen, daß die dort erfolgende Definition der Einfluß-, Wirkungs- und Operationsbereiche von Transaktionen für konkrete Netze nicht mehr aufgeführt werden muß. Ebenso läßt sich auf die dort definierten Anwendungen von Variablenbindungsfunktionen auf Familien von Multimengen atomarer Formelvorkommnisse verzichten. Beide Aspekte sind für alle konkreten Netze in derselben Weise definiert.

120) Aufgrund dieser impliziten Vereinbarung verweisen alle konkreten Netze auf das zuvor erläuterte abstrakte Definitionsschema Synthetischer Netze. In diesem Schema sind alle erforderlichen Definitionskonstituenten explizit definiert. Daher trifft hier die Vorhaltung einer unvollständigen Explizierung nicht zu, die in bezug auf Stelle/Transition-Netze an früherer Stelle thematisiert worden war.

5.1.2.2 Eine spezielle Darstellungsweise

Das allgemeine Definitionsschema $SN=(TOP,SPEC_{MSIG};BES,M_0;IB)$ für Synthetische Netze fällt - im Vergleich zu sonst üblichen Netzdefinitionen - relativ kompliziert aus. Diese Konsequenz läßt sich nicht vermeiden, sofern die universelle Definitionsgültigkeit und die explizite Darlegung aller Definitionskonstituenten[1)] angestrebt wird. Um den Umgang mit konkreten Synthetischen Netzen zu erleichtern, wird eine spezielle Darstellungsweise eingeführt. Sie zeichnet sich durch ihre Kompaktheit und Transparenz aus. Zunächst werden Redundanzen im allgemeingültigen Definitionsschema[2)] ausgenutzt, um durch ihre Eliminierung zu einer verdichteten Netzdarstellung zu gelangen. Darüber hinaus werden Vereinfachungen vorgenommen. Sie lassen sich im Hinblick auf die spätere Implementierung von Synthetischen Netzen mit Hilfe des Softwarepakets PASIPP auf der Basis der Programmiersprache PROLOG rechtfertigen[3)]. Diese beiden kompaktifizierenden Aspekte werden durch die graphische Repräsentation von Netzbestandteilen ergänzt. Die visualisierte Aufbereitung der graphischen Komponente sorgt für die Transparenz der Netzdarstellung.

Darstellungsweise Synthetischer Netze

Ein Synthetisches Netz $SN=(TOP,SPEC_{MSIG};BES,M_0;IB)$ wird in semi-graphischer Weise dargestellt als die Kombination aus:

- ❑ einer graphischen Repräsentation der Netztopologie TOP sowie "wesentlicher" Aspekte von Netzbeschriftung BES und Netzmarkierung M_0 (Netzgraphik);
- ❑ einer nicht-graphischen Legende, welche die Beschriftung und Markierung der graphischen Netzdarstellung durch Komponenten aus der Netzspezifikation $SPEC_{MSIG}$ erklärt (Netzlegende).

Die Integritätsbedingungen aus der Menge IB gehen in die semi-graphische Netzdarstellung nicht ein[4)].

<u>Erläuterungen und Ergänzungen zur Darstellungsweise Synthetischer Netze:</u>

a) Die Netzgraphik stellt einen beschrifteten bipartiten gerichteten Graphen dar. Dieser Graph wird aber nicht in seiner mathematisch definierten Form, sondern stets in seiner visualisierten Gestalt dargestellt. Für die Netzgraphik gelten folgende Vereinbarungen:

- ❑ Jede Transition wird durch ein Rechteck repräsentiert. Als Standardfall dient ein quadratisches Rechteck. Es wird mit dem Symbol "t_n" der dargestellten Transition beschriftet.
- ❑ Jede Transition t_n läßt sich zusätzlich mit dem Namen tr_v derjenigen Transaktion beschriften, die der Transition durch $btt(t_n)=tr_v$ zugewiesen ist[5)].
- ❑ Jede Stelle wird durch einen Kreis dargestellt. Er wird mit dem Symbol "s_m" der repräsentierten Stelle beschriftet.
- ❑ Jede Stelle s_m darf als zusätzliche Anschrift den Namen $Prä_u$ desjenigen Prädikatssymbols tragen, das der Stelle durch $bsp(s_m)=Prä_u$ zugeordnet ist.
- ❑ Jede Einflußkante (s_m,t_n) einer Transition t_n wird durch einen Pfeil repräsentiert, der vom Kreis der Stelle s_m zum Rechteck der Transition t_n gerichtet ist.

- Eingangs- und Informationskanten werden dadurch unterschieden, daß erste durch einen ununterbrochenen, letzte dagegen durch einen unterbrochenen Pfeil dargestellt werden.
- Jede Ausgangskante (t_n,s_m) einer Transition t_n wird durch einen ununterbrochenen Pfeil repräsentiert, die vom Rechteck der Transition t_n zum Kreis der Stelle s_m gerichtet ist.
- Jede Stelle s_m kann mit ihrer Markierung $M_0(s_m)$ beschriftet werden, die ihr unter der Ausgangsmarkierung M_0 im zugrundeliegenden Netz SN zukommt.
- Alle Kopien der Basismarke mit s=0 werden als "m_0" oder "$\emptyset$" dargestellt[6)].
- Verschiedene Kopien m_s derselben strukturierten Markenart "s" mit $s \in \{1,..,A+B\}$ werden durch einen differenzierenden Index "d" unterschieden, der für das gesamte Netz gilt[7)]. Die Markenkopien werden dann als "$m_{s.d}$" dargestellt.
- Für die markierungsbezogene Stellenbeschriftung gilt eine vereinfachte Notation: Falls eine Stelle s_m mit zugeordneten Prädikatssymbol $Prä_u$ die Ausgangsmarkierung $M_0(s_m) = FAK_{u.0}$ besitzt und das Fakt $fakt_0(mu_{u.0.d}, prä_u(m_{s(u.1).d},...,m_{s(u.Ku).d})) \in FAK_{u.0}$ in der Markierung der Stelle s_m $mu_{u.0.d}$-fach enthalten ist mit $mu_{u.0.d} \in \mathcal{N}_+$, dann wird das Markentupel $<m_{s(u.1).d},..., m_{s(u.Ku).d}>$ unter Vernachlässigung des Prädikatssymbolnamens "$Prä_u$"[8)] und der Faktbezeichnung "$fakt_0$" insgesamt $mu_{u.0.d}$-mal in oder neben den Kreis geschrieben, der die Stelle s_m repräsentiert. Wenn eine Stelle s_m wegen $M_0(s_m) = \emptyset$ auf die leere Faktenmenge abgebildet wird, besitzt der stellenrepräsentierende Kreis keine markierungsbezogene Anschrift.
- Die innere Struktur der Kopien $m_{s(u.k).d}$ von strukturierten Marken mit $k \in \{1,...,K_u\}$ aus den Markentupeln $<m_{s(u.1).d},...,m_{s(u.Ku).d}>$ wird in der Netzgraphik nicht aufgelöst, sondern erst in der Netzlegende erklärt[9)]. Daher wird von einer impliziten Kurznotation der Markenkopien $m_{s(u.k).d}$ gesprochen.
- Kopien der unstrukturierten Basismarke werden in der Regel durch ihr Symbol "$\emptyset$" dargestellt[10)]. Ebenso können sie durch den Ausdruck "m_0" vertreten werden[11)].
- Jede Kante, die zwischen einer Transition t_n mit zugehöriger Transaktion tr_v und einer Stelle s_m mit zugehörigem Prädikatssymbol $Prä_u$ verläuft, kann mit demjenigen Kantengewicht $MTAV_{u.v}$, $MTAI_{u.v}$ oder $MTAN_{u.v}$ beschriftet werden, die dieser Kante im zugrundeliegenden Netz SN zugeordnet ist.
- Falls die Kanten mit ihren Gewichten beschriftet werden, erfolgt eine ähnliche Vereinfachung wie für die markierungsbezogene Stellenbeschriftung: Die Multimengen der Kantengewichte $MTAV_{u.v}$, $MTAI_{u.v}$ oder $MTAN_{u.v}$ werden zunächst als maximal vereinfachte formale Summen dargestellt[12)]. Die Teilausdrücke $mu_{u.n.d} \bullet prä_u(m_{s(u.1).d},...,m_{s(u.Ku).d})$, die diese formalen Summen konstituieren, werden unter Vernachlässigung des Prädikatsnamens "$prä_u$" durch die kompaktere Notation $mu_{u.n.d} \bullet <m_{s(u.1).d},...,m_{s(u.Ku).d}>$ ersetzt[13)]. Daher wird jedes Kantengewicht als eine formale Summe aus Markentupeln $<m_{s(u.1).d},...,m_{s(u.Ku).d}>$ repräsentiert.
- Die Kopien $m_{s(u.k).d}$ von strukturierten Marken, die mit $k \in \{1,...,K_u\}$ in den Kantengewichten enthalten sind, werden hinsichtlich ihrer inneren Struktur wiederum nicht expliziert (implizite Kurznotation). Die differenzierenden Indizes "d" der Kopien $m_{s(u.k).d}$ von strukturierten Marken werden ausgelassen, falls die innere Markenstruktur mindestens eine Variable enthält. Denn dann können die variablen Kantenanschriften $m_{s(u.k)}$ unter verschiedenen Variablenbelegungen durch jeweils unterschiedliche Markenkopien $m_{s(u.k).d}$ gebunden werden[14)].
- Jede Transition t_n *kann* mit Restriktionsformeln for_z aus der Formelmenge RES_v beschriftet werden, die zur Deklaration ihrer zugeordneten Transaktion tr_v gehört[15)]. Falls diese Beschriftungsoption ausgeübt wird, werden jedoch nur diejenigen Restriktionsformeln als Transitionsanschriften in Betracht gezogen, die Haupttestbedingungen oder Bestimmungsgleichungen ausdrücken[16)]. Falls eine Transition sowohl mit mindestens einer Haupttest-

bedingung als auch mit mindestens einer Bestimmungsgleichung beschriftet wird, so lassen sich die Haupttestbedingung(en) und Bestimmungsgleichung(en) durch eine unterbrochene horizontale Linie voneinander separieren. Dadurch tritt ihre unterschiedliche Restriktionsqualität besonders deutlich hervor[17)].

b) Die Netzlegende wird in einer Form notiert, die an die algebraische Darstellungsweise des Signaturkonzepts angelehnt ist. Sie besteht aus vier Sektionen für die Deklaration der Marken, Prädikatssymbole, Transaktionen und Faktenmengen sowie für deren Zuordnungen zu den topologischen Konstrukten aus der graphischen Netzdarstellung:

<u>Netzlegende:</u>

<u>Marken/Operationssymbole:</u>

$attribut_1$: $OB_{att.1}$ / $= Att_{j(1)}(attribut_{q(j(1).1)} \ldots attribut_{q(j(1).Kj(1))})$ [;...]

...

$attribut_Q$: $OB_{att.Q}$ / $= Att_{j(Q)}(attribut_{q(j(Q).1)} \ldots attribut_{q(j(Q).Kj(Q))})$ [;...]

«m_0» ≈ bas_marke: SYMBOL

«$m_{j(1)}$» ≈ $att\_marke_1 = Marke_{j(1)}(attribut_{q(j(1).1)} \ldots attribut_{q(j(1).Kj(1))})$ [;...]

...

«$m_{j(A)}$» ≈ $att\_marke_A = Marke_{j(A)}(attribut_{q(j(A).1)} \ldots attribut_{q(j(A).Kj(A))})$ [;...]

«$m_{j(A+1)}$» ≈ $str\_marke_{A+1}=Struk_{j(A+1)}(str\_marke_{s(j(A+1).1)} \ldots str\_marke_{s(j(A+1).Kj(A+1))})$ [;...]

...

«$m_{j(A+B)}$» ≈ $str\_marke_{A+B} = Struk_{j(A+B)}(str\_marke_{s(j(A+B).1)} \ldots str\_marke_{s(j(A+B).Kj(A+B))})$ [;...]

<u>Stellen/Prädikatssymbole:</u>

s_1: $Prä_1(sor\_marke_{s(1.1)} \ldots sor\_marke_{s(1.K1)})$
[$markenkapazität_1$=KAP_1]

...

s_M: $Prä_M(sor\_marke_{s(M.1)} \ldots sor\_marke_{s(M.KM)})$
[$markenkapazität_M$=KAP_M]

<u>Transitionen/Transaktionen:</u>

t_1: tr_1

$$\Sigma(d\in DKV_{u.1}): mu_{u.1.d} \bullet <m_{s(u.1).d},...,m_{s(u.Ku).d}>$$
$$\approx \Sigma(d\in DKV_{u.1}): mu_{u.1.d} \bullet prä_u(m_{s(u.1).d}{}^{ex},...,m_{s(u.Ku).d}{}^{ex}) \quad \text{für alle } u\in IVB(tr_1)$$

$$\Sigma(d\in DKI_{u.1}): mu_{u.1.d} \cdot prä_u<m_{s(u.1).d},...,m_{s(u.Ku).d}>$$
$$\approx \Sigma(d\in DKI_{u.1}): mu_{u.1.d} \bullet (m_{s(u.1).d}{}^{ex},...,m_{s(u.Ku).d}{}^{ex}) \quad \text{für alle } u\in IIB(tr_1)$$

$$for_{z(1.1)},...,for_{z(1.H1)}$$

$$\Sigma(d\in DKN_{u.1}): mu_{u.1.d} \bullet <m_{s(u.1).d},...,m_{s(u.Ku).d}>$$
$$\approx \Sigma(d\in DKN_{u.1}): mu_{u.1.d} \bullet prä_u(m_{s(u.1).d}{}^{ex},...,m_{s(u.Ku).d}{}^{ex}) \quad \text{für alle } u\in INB(tr_1)$$

...

t_N: tr_N

$$\Sigma(d\in DKV_{u.N}): mu_{u.N.d} \bullet <m_{s(u.1).d},...,m_{s(u.Ku).d}>$$
$$\approx \Sigma(d\in DKV_{u.N}): mu_{u.N.d} \bullet prä_u(m_{s(u.1).d}{}^{ex},...,m_{s(u.Ku).d}{}^{ex}) \quad \text{für alle } u\in IVB(tr_N)$$

$$\Sigma(d\in DKI_{u.N}): mu_{u.N.d} \bullet <m_{s(u.1).d},...,m_{s(u.Ku).d}>$$
$$\approx \Sigma(d\in DKI_{u.N}): mu_{u.N.d} \bullet prä_u(m_{s(u.1).d}{}^{ex},...,m_{s(u.Ku).d}{}^{ex}) \quad \text{für alle } u\in IIB(tr_N)$$

$$for_{z(N.1)},...,for_{z(N.HN)}$$

$$\Sigma(d\in DKN_{u.N}): mu_{u.N.d} \bullet <m_{s(u.1).d},...,m_{s(u.Ku).d}>$$
$$\approx \Sigma(d\in DKN_{u.N}): mu_{u.N.d} \bullet prä_u(m_{s(u.1).d}{}^{ex},...,m_{s(u.Ku).d}{}^{ex}) \quad \text{für alle } u\in INB(tr_N)$$

<u>Fakten:</u>

$$mu_{1.0.d} \bullet <m_{s(1.1).d},...,m_{s(1.K1).d}> \approx fakt_0(mu_{1.0.d}, prä_1(m_{s(1.1).d}{}^{ex},...,m_{s(1.K1).d}{}^{ex})) \quad \text{für alle } d\in DF_1$$

...

$$mu_{u.0.d} \bullet <m_{s(u.1).d},...,m_{s(u.Ku).d}> \approx fakt_0(mu_{u.0.d}, prä_u(m_{s(u.1).d}{}^{ex},...,m_{s(u.Ku).d}{}^{ex})) \quad \text{für alle } d\in DF_u$$

...

$$mu_{U.0.d} \bullet <m_{s(U.1).d},...,m_{s(U.KU).d}> \approx fakt_0(mu_{U.0.d}, prä_U(m_{s(U.1).d}{}^{ex},...,m_{s(U.KU).d}{}^{ex})) \quad \text{für alle } d\in DF_U$$

Der Netzlegende liegen folgende Vereinbarungen zugrunde:

- Für Stellen s_m und zugeordnete Prädikatssymbole $Prä_u$ sowie für Transitionen und zugeordnete Transaktionen tr_v werden jeweils identische Indexierungen vorgenommen:

  $bsp(s_m) = Prä_u \rightarrow m=u$ für alle $s_m \in S$ und $Prä_u \in PRÄ$

  $btt(t_n) = tr_v \rightarrow n=v$ für alle $t_n \in T$ und $tr_v \in TR$

- Die Indexmengen DM_s für $s\in\{1,...,A+B\}$, die Indexmengen $DKV_{u.n}$, $DKI_{u.n}$ und $DKN_{u.n}$ für $u\in IVB(tr_n)$, $u\in IIB(tr_n)$ bzw. $u\in INB(tr_n)$ und $n\in\{1,...,N\}$ sowie die Indexmengen DF_u für $u\in\{1,...,U\}$ stellen marken-, kanten- bzw. faktenbezogene Mengen von differenzierenden Indizes "d" dar. Sie dienen dazu, verschiedene Kopien $m_{s.d}$ derselben Marke sor_marke$_s$, unterschiedliche Argumente $(m_{s(u.1).d},...,m_{s(u.Ku).d})$ derselben atomaren Formel $prä_u$ in einem Kantengewicht[18] bzw. verschiedene Argumente $(m_{s(u.1).d},...,m_{s(u.Ku).d})$ derselben atomaren Formel $prä_u$ in ihrer Ausgangsfaktenmenge auseinanderzuhalten. Damit die Indizes ihre differenzierende Funktion erfüllen, werden sie so gewählt, daß die Gesamtindizes aller indizierten Ausdrücke im gesamten Netz paarweise verschieden sind[19]. Falls eine Indexmenge nur genau einen Index umfaßt, kann dieser auch vernachlässigt werden[20].

❑ Kantengewichte und Faktenmengen stellen jeweils Multimengen aus teilevaluierten bzw. konstanten atomaren prädikatenlogischen Formeln dar. Ihre Argumente werden in der Netzlegende grundsätzlich mit vollständig explizierten[21] Termen notiert[22]. Die Terme werden also als Kombinationen aus Operatoren, Variablen und atomaren formalen Objekten (Konstanten) dargestellt. Bei den Operatoren kann es sich um die Operatoren für die Strukturierung von Kompositmarkenkopien, für die Erzeugung von Attributmarkenkopien oder für Gestaltung zusammengesetzter Attributausprägungen handeln. Die Variablen und Konstanten vertreten bzw. sind per constructionem stets Ausprägungen von originär definierten, atomaren Attributen.

❑ Falls ein vollständig explizierter Basisterm im Argument eines Kantengewichts für die aktuell behandelte Modellierungsaufgabe keine Relevanz besitzt, wird die anonyme Variable "_" verwendet. Sie wurde bereits früher eingeführt. Hierdurch gehen keine Informationen verloren, sondern werden nur als irrelevant ausgeblendet[23]. Fakten müssen immer eindeutig bestimmt sein. Daher kommt für ihre Formulierung die anonyme Variable nicht in Betracht.

❑ Die Argumente der Prädikatssymbole werden nicht vollständig expliziert. Als Argumentkomponenten werden jeweils die Markensorten aufgeführt, über denen das Prädikatssymbol definiert ist. In der Sektion "Marken/Operationssymbole" sind diese Markensorten als Zielsorten von Operationssymbolen für die Erzeugung von Attributmarken oder für die Strukturierung von Kompositmarken erklärt. Die Sorten in den Argumenten dieser Operationssymbole können ihrerseits als Zielsorten von anderen Operationssymbolen erklärt sein usw. Die gesamte Information über die sortierte Argumentstruktur der Prädikatssymbole läßt sich aus der Sektion "Marken/Operationssymbole" indirekt erschließen.

❑ In der Sektion "Marken" werden nur diejenigen Attribut- und Markensorten sowie Objektmengen explizit aufgeführt, die zur Definition aller sortierten Markenarten "s" aus der netzspezifischen Markenartenmenge $SMA=\{s: s=0,1,...,S\}$ mit $S=A+B$ und ihrer zulässigen Kopien unbedingt erforderlich sind. Alle übrigen Attribut- und Markendeklarationen werden implizit in der Weise vorausgesetzt, wie sie im allgemeinen Definitionsschema für Synthetische Netze festgelegt wurden[24].

❑ Der Definitionsbereich $OB_{att.q}$ einer originär definierten Attributsorte $attribut_q$ ist immer eine von den bereits eingeführten allgemeinen Objektmengen[25].

❑ Falls eine spezielle Objektmenge $OB_{att.q}=\{ob_1,...,ob_{Kq}\}$ so spezifiziert werden soll, daß nur die formalen Objekte (Konstanten) $ob_1,...,ob_{Kq}$ als zulässige Attributausprägungen in Betracht kommen, dann werden die hierfür erforderlichen 0-stelligen Operationssymbole nicht explizit dargestellt. Statt dessen wird die Kurznotation $ob_1,...,ob_{Kq}: \rightarrow OB_{att.q}$ verwendet. Sie wird unmittelbar an die allgemeine Objektmenge der Attributsorte angeschlossen.

❑ Die Notation "/" bei der Definition von Attributsorten drückt aus, daß eine Attributsorte $attribut_q$ entweder originär oder aber derivativ definiert ist. Im ersten Fall wird für die originäre Attributsorte der zugehörige Definitionsbereich $OB_{att.q}$ zulässiger Attributausprägungen angegeben. Im zweiten Fall wird die derivative Attributsorte als Zielsorte eines Operationssymbols "Att_q" dargestellt, mit dessen Hilfe die Attributsorte aus anderen Attributsorten aufgebaut wird. Die Definitionsbereiche der derivativ definierten Attributsorten werden nicht expliziert[26].

❑ Die Notation "[;...]" am Ende der Definitionen von Attribut- und von Markensorten weist auf die Möglichkeit hin, daß die Sorten jeweils als die Zielsorte mehrerer Operationssymbole definiert sein können. Falls eine solche multiple Zielsortendefinition vorliegt, werden die Applikationen der beteiligten Operationssymbole auf ihre Argumente nacheinander aufgelistet und jeweils durch ein ";" voneinander getrennt[27]. Falls eine Attribut- oder Markensorte einfach definiert ist, entfällt die Notation"[;...]" am Definitionsende[28].

- Die Notation "«m_0»" ordnet der Basismarke, die in der Netzlegende durch die Markensorte "bas_marke" definiert ist, die Darstellung "m_0" für ihre Kopien in der Netzgraphik zu[29]. Analog ordnen die Notationen "«m_a» ≈" und "«m_{A+b}» ≈" jeder Marke, die in der Netzlegende durch die Markensorte "att_marke$_a$" mit $a \in \{1,..,A\}$ als Attributmarke bzw. durch die Markensorte "str_marke$_b$" mit $b \in \{1,..,B\}$ als Kompositmarke definiert ist, die Bezeichnungen "m_a" bzw. "m_{A+b}" für ihre Kopien in der Netzgraphik eineindeutig zu[30].

- In der Sektion "Stellen/Prädikatssymbole" wird für jede Stelle s_m ihr zugeordnetes Prädikatssymbol Prä$_u$ mit u=m angeführt. Die Argumente der Prädikatssymbole werden grundsätzlich aus Markensorten formiert. Die Markenkapazitäten KAP_m der Stellen s_m werden nur dann explizit durch Kapazitätsgleichungen der Form "markenkapazität$_m$=KAP_m" spezifiziert, wenn sie endlich sind. Für alle nicht notierten Markenkapazitäten gilt die implizite Voreinstellung $KAP_m = \omega$.

- Die Argumente der Prädikatssymbole können nur für ein konkret vorliegendes Synthetisches Netz unmittelbar durch die Markensorten aus der Sektion "Marken/Operationssymbole" spezifiziert werden. In der o.a. abstrakten Beschreibung der Netzdarstellung bleibt es jedoch unbekannt, durch welche Markensorten die Argumentstellen der Prädikatssymbole jeweils belegt werden. Daher dienen die Markensorten "sor_marke" nur als Platzhalter. Sie werden für jedes konkrete Netz entweder durch die Markensorten "att_marke" oder "str_marke", die in der Sektion "Marken/Operationssymbole" explizit eingeführt worden sind, oder durch die implizit definierte Markensorte "bas_marke" ersetzt.

- Die Variablen und die Terme aus der zugrundeliegenden MSIG-Spezifikation werden in der Netzlegende nicht explizit aufgeführt. Statt dessen gelten sie als implizit vereinbart.

- In der Sektion "Transitionen/Transaktionen" wird für jede Transition t_n ihre zugeordnete Transaktion tr_v mit v=n angeführt. Darauf folgen die Kantengewichte $MTAV_{u.v}$, $MTAI_{u.v}$ und $MTAN_{u.v}$ für den Vor-, Informations- bzw. Nachbereich der Transaktion tr_v. Die Gruppierung der Kantengewichte hinsichtlich der Bereichszugehörigkeit ihrer zugehörigen Prädikatssymbole ergibt sich eindeutig aus der unterschiedlichen Darstellungsweise der drei Kantenkategorien in der Netzgraphik. Die Kantengruppierung gestattet es, Eingangs-, Informations- und Ausgangskanten zu unterscheiden.

- Die Kantengewichte stellen Multimengen aus teilevaluierten atomaren Formeln prä$_u$(..., $m_{s(u.k).d}^{ex}$,...) dar. Die Multimengen werden als maximal vereinfachte formale Summen notiert. Die darin vorkommenden Markenkopien $m_{s(u.k).d}^{ex}$ werden jeweils hinsichtlich ihrer internen Zusammensetzung vollständig expliziert[31], sofern es sich nicht um Kopien der strukturlosen Basismarke handelt. Dadurch werden die Markenkopien $m_{s(u.k).d}$ aus den Formelargumenten, die in der Netzgraphik nur in impliziter Kurznotation dargestellt wurden, nunmehr in der Netzlegende vollständig spezifiziert. Das Symbol "≈" stellt die eineindeutige Zuordnung zwischen der impliziten Kurznotation aus der Netzgraphik und der vollständigen Explizierung in der Netzlegende her[32]. Zwecks Erhöhung der Übersichtlichkeit wird zunächst die implizite Kurznotation und erst danach die vollständige Explizierung aufgeführt. Darüber hinaus wird bei der impliziten Kurznotation die Kurzschreibweise "<...,$m_{s(u.k).d}$,...>" genutzt, die es gestattet, auf den jeweils zugehörigen Formelnamen "prä$_u$" zu verzichten.

- Zusätzlich werden in der Sektion "Transitionen/Transaktionen" für jede Transition t_n Restriktionsformeln $for_{z(n.h)}$ mit $h \in \{1,...,H_n\}$ aufgelistet. Es handelt sich genau um diejenigen Formeln, mit deren Hilfe Bestimmungsgleichungen oder Haupttestbedingungen für die Übergangsoperation der zugeordneten Transaktion tr_v mit v=n ausgedrückt werden und die in der zugehörigen Formelmenge RES_v des zugrundeliegenden Synthetischen Netzes explizit enthalten sind.

- In der Sektion "Fakten" wird ein multiples Fakt $fakt_0(mu_{u.0.d}, prä_u(m_{s(1).d}^{ex},...,m_{s(Ku).d}^{ex}))$ in der gleichen Weise notiert, wie es zuvor für die Formeln aus den Kantengewichten erläutert wurde. Wiederum wird der vollständig explizierten Faktendarstellung die implizite Kurznotation $mu_{u.0.d} \bullet <m_{s(1).d},...,m_{s(Ku).d}>$ aus der Netzgraphik eineindeutig zugeordnet[33].
- Ein Fakt des Prädikatssymbols $Prä_u$ mit $d \in \{1,...,D_u\}$ wird genau dann aufgelistet, wenn $fakt_0(mu_{u.0.d}, prä_u(m_{s(1).d},...,m_{s(Ku).d})) \in FAK_{u.0}$ gilt. Wegen der vorausgesetzten Endlichkeit aller prädikatspezifischen Faktenmengen kann die gesamte Ausgangsmarkierung M_0 eines Synthetischen Netzes in der Sektion "Fakten" vollständig dargestellt werden[34].

c) Netzgraphik und -legende besitzen komplementären Charakter. Die Netzgraphik repräsentiert erstens die gesamte Netztopologie. Dies liegt nahe, weil die Netztopologie TOP einen mathematisch definierten Graphen darstellt, der sich in "natürlicher Weise" graphisch repräsentieren läßt. Zweitens dient die Netzgraphik der Hervorhebung aller nicht-topologischen Aspekte eines Synthetischen Netzes, die im jeweils ausgeführten Argumentationskontext besonders hervorgehoben werden sollen[35] und in ihrer Darstellungskomplexität zu keiner perzeptiven Überfrachtung[36] der Netzgraphik führen. Damit wird die kognitionspsychologische Erfahrung gewürdigt, daß sich graphische Repräsentationsweisen empfehlen, um Sachverhalte mittleren Komplexionsniveaus[37] zu veranschaulichen. Die Netzlegende übernimmt dagegen den komplementären Part, jene nicht-topologischen Netzaspekte darzustellen, die von der Netzgraphik nicht abgedeckt werden. Dabei kann es sich einerseits um jene Aspekte handeln, die im aktuellen Argumentationskontext nicht von herausragender Bedeutung sind. Andererseits können Aspekte betroffen sein, deren graphische Repräsentation so komplex ausfiele, daß sie sich infolge perzeptiver Überfrachtung nicht mehr rechtfertigen ließe[38]. Beide vorgenannten Fälle können auch kombiniert auftreten.

d) Die Rechtecke, Kreise und Pfeile, die Transitionen, Stellen bzw. Kanten repräsentieren, können vereinfacht als Transitionen, Stellen bzw. Kanten angesprochen werden[39].

e) Die Stellen-, Kanten- und Transitionsbeschriftungen mit Markierungen, Formelmultimengen bzw. Restriktionsformeln müssen mindestens einmal notiert werden. Es besteht Wahlfreiheit, ob dies entweder in der Netzgraphik oder aber in der Netzlegende erfolgt. Auch die - allerdings redundante - Beschriftungsexplikation in beiden Darstellungskomponenten ist zulässig[40]. Im Regelfall wird vom Verf. eine redundanzfreie, aber gespaltene Darstellungsweise bevorzugt: Diejenigen Beschriftungsteile, die im jeweils aktuellen Argumentationskontext eine besondere Rolle spielen, werden in der Netzgraphik angeführt, alle übrigen dagegen in der Netzlegende.

f) Die Sektion "Fakten" repräsentiert die Ausgangsmarkierung M_0 durch ihre Faktenmenge FAK_0. In der Fakten-Sektion werden alle Fakten $fakt_0(mu_{u.0}, prä_u(m_{s(u.1)},...,m_{s(u.Ku)})$ aufgelistet, welche die Prädikatssymbole $Prä_u$ von Stellen s_m im originären Netzzustand "r=0" erfüllen. Durch die Komponenten "$prä_u$" sind die Fakten jeweils genau einer Stelle s_m mit $bsp(s_m) = Prä_u$ eindeutig zugeordnet. Daher braucht die Faktenauflistung - im Gegensatz zur Darstellung des Markierungsvektors $\underline{M}_0 = (M_0(s_1),...,M_0(s_M))$ - nicht hinsichtlich der involvierten Stellen gesondert strukturiert zu werden. Dies rechtfertigt nachträglich nochmals die frühere Vereinbarung, die Ausgangsfaktenmenge FAK_0 eines Synthetischen Netzes nicht nur als Familie ($FAK_{u.0}$: $u = 1,...,U$) der stellenspezifischen Faktenmengen $FAK_{u.0}$ mit $FAK_{u.0} = M_0(s_m)$ und $bsp(s_m) = Prä_u$ für alle Prädikatssymbole $Prä_u$ und Stellen s_m aufzufassen. Statt dessen kann die Ausgangsfaktenmenge wegen der konstruktionsbedingten Disjunktheit der stellenspezifischen Faktenmengen ebenso als Vereinigungsmenge $\cup(u \in \{1,...,U\})$: $FAK_{u.0}$ behandelt werden.

g) Die Sektion "Fakten" kann auch dazu benutzt werden, um die Faktenmenge FAK_r für einen beliebigen Netzzustand "r" zu repräsentieren[41]. Alle Vereinbarungen, die für die Darstellung der Ausgangsmarkierung getroffen wurden gelten analog - bis auf die Ersetzung des Indexes "0" durch den Index "r". Aus der Indexierung der Fakten, die in dieser Sektion aufgelistet sind, läßt sich der jeweils betrachtete Netzzustand ablesen[42].

h) In der Sektion "Fakten" wird im Regelfall nur die Faktenmenge FAK_r für genau einen Netzzustand aufgeführt. Daher besitzt der Index "r" der faktischen Formeln $fakt_r(mu_{u.r}, prä_u(m_{s(u.1)}, \ldots, m_{s(u.Ku)})$ keinen unterscheidenden, sondern nur identifizierenden Charakter: Er gibt den Netzzustand "r" - die Netzmarkierung M_r - an, auf den bzw. die sich *alle* faktischen Formeln aus der Faktenmenge FAK_r beziehen[43]. Dieser Netzzustand kann aber beliebig gewählt sein[44]. Falls Faktenmengen für mehrere alternative Netzmarkierungen angegeben werden sollen, muß die Sektion "Fakten" in entsprechend viele Subsektionen aufgespalten werden. In jeder Subsektion wird dann die gesamte Faktenmenge des subsektionsspezifischen Netzzustands "r" ausgewiesen[45]. Darauf wird aber nur selten zurückgegriffen, weil diese Darstellungsweise aufwendig und unübersichtlich ist[46].

i) Die vier Sektionen "Marken/Operationssymbole", "Stellen/Prädikatssymbole", "Transitionen /Transaktionen" und "Fakten" korrespondieren - abgesehen von einigen geringfügigen technischen Abweichungen[47] - mit charakteristischen Deklarationsbereichen[48] von Turbo-PROLOG-Programmen, die später der Netzimplementierung zugrundegelegt werden. Diese Übereinstimmung wurde bewußt konstruiert, um die intendierte Netzimplementierung auf PROLOG-Basis vorzubereiten.

Ein Beispiel zur speziellen Netzdarstellung:

Es wird ein bewußt einfach gehaltenes Beispiel vorgelegt, um die wesentlichen Darstellungskomponenten transparent hervortreten zu lassen[49]. Das Exempel ist einer konventionellen Darstellung von Prädikat/Transition-Netzen entnommen[50]. Es betrifft die grobe Modellierung von Aktionen, bei deren Vollzug jeweils ein Mietvertrag mit einem Kunden über die Nutzung eines Mietwagens abgeschlossen wird. Diese gleichartigen Aktionen werden durch eine Transaktion tr_1 = "vertragsabschluss" abgebildet. Zwei jeweils 1-stellige Prädikatssymbole mit den Namen "Mietwilliger_kunde" und "Verfuegbares_auto" geben durch ihre Extensionen an, welche Kunden einen Mietvertrag begehren und welche Mietwagen hierfür zur Verfügung stehen. Ein 2-stelliges Prädikatssymbol mit dem Namen "Kunde_mit_vertrag" drückt aus, daß einem Kunden nach Vertragsabschluß ein Mietwagen zugeordnet ist. Ein weiteres 1-stelliges Prädikatssymbol mit dem Namen "Verliehenes_auto" führt in seiner Extension jene Mietwagen auf, die zwar existieren, aber aufgrund früherer Vertragsabschlüsse vorläufig nicht mehr vermietet werden können. Jedes Ausführen der Transaktion "vertragsabschluss" bedeutet, daß ein Mietvertrag mit einem Kunden über einen Mietwagen abgeschlossen wird. Dabei wird jeweils die Haupttestbedingung überprüft, ob ein mietwillige Kunde mindestens 21 Jahre alt ist.

In Übereinstimmung mit dem Original werden alle Beschriftungskomponenten, die Variablen darstellen, mit Großbuchstaben begonnen. Dies gilt für die teilevaluierten atomaren Formeln der Kantenanschriften und die eine Haupttestbedingung der Transitionsanschrift. Aus mnemotechnischen Erwägungen wird die Variable "X" einer Sorte "sort" als "Sort" notiert. Attributausprägungen, die Zeichenfolgen der globalen Sorte "string" darstellen, werden jeweils von einem Begrenzungszeichen (") beidseitig eingeschlossen[51]. Alle involvierten Marken stellen Attributmarken dar. Die zugehörigen Attributbezeichnungen wurden aus dem Original im Interesse einer unmittelbaren Vergleichsmöglichkeit unverändert übernommen. Nur das Attribut "adressfeld" ist derivativ, alle anderen Attribute sind dagegen originär definiert. Bei drei Prädikatssymbolen handelt es sich jeweils um 1-stellige Prädikatssymbole, bei einem um ein 2-stelliges Prädikatssymbol. Markenkopien werden in der Netzgraphik in der kompakten, impliziten Form "m_K" für Kopien der Kundenmarke und "m_A" für Kopien der Automarke repräsentiert.

Gegenüber dem Prädikat/Transition-Netz des Originals wurden zwei wesentliche Veränderungen vorgenommen. Erstens nimmt das Original explizit auf Marken im hier eingeführten Verständnis überhaupt keinen Bezug[52)]. Dies ist für Prädikat/Transition-Netze typisch. In der Darstellungsweise Synthetischer Netze werden jedoch die beiden selbständig existenzfähigen Objekte "Kunde" und "Auto" nach Maßgabe der früher entwickelten Marken-Ontologie als Marken modelliert. Dadurch wird ein Informationsverlust über die Attribute "typ" und "baujahr" des jeweils gemieteten Wagens vermieden, der im Original eintritt[53)]. Zweitens wird die undifferenzierte Attributaufzählung in den Kantenanschriften des Originals hier durch eine hierarchische Strukturierung ersetzt. Mit ihrer Hilfe werden die Attribute, die jeweils Eigenschaften eines der beiden selbständig existenzfähigen Objekte darstellen, derjenigen Marke untergeordnet, die das betroffene Objekt modelliert.

Abb. 51 auf der nächsten Seite enthält das beispielabbildende Synthetische Netz als Kombination aus einer Netzgraphik und ihrer Legende. Dabei wird auf den Ausgangszustand "r=0" des Netzmodells mit der Netzmarkierung $M_0 = Fak_0$ Bezug genommen.

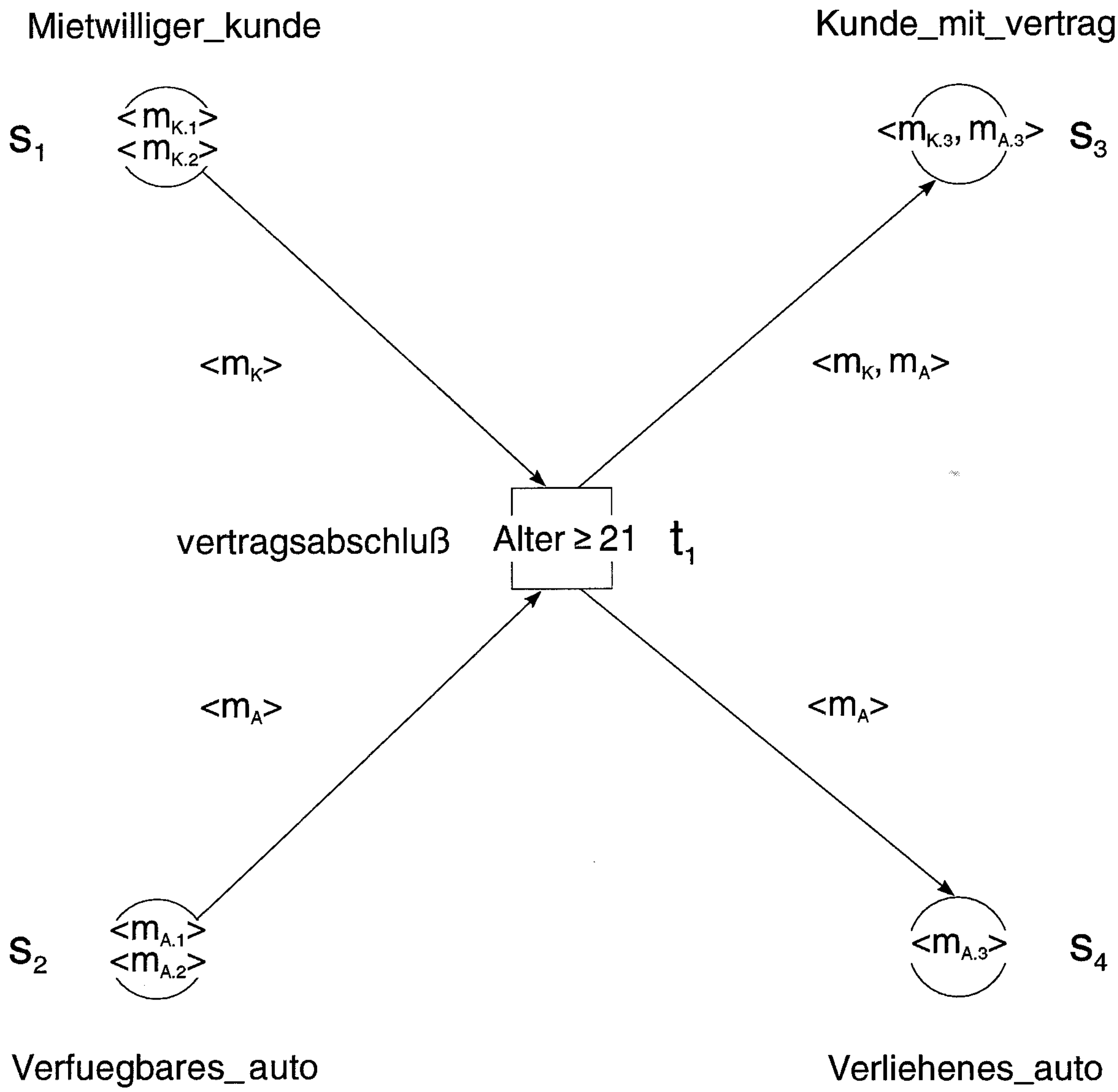

Abb. 51: Beispielnetz einer Autovermietung für den Ausgangszustand "r=0"

Netzlegende:

Marken/Operationssymbole:

alter: INTEGER
baujahr: INTEGER
kfz_nr: STRING
name: STRING
strasse: STRING
typ: STRING
wohnort: STRING
adressfeld = Adresse(strasse wohnort)
«m_K» ≈ kunde = Kundenmarke(name adressfeld alter)
«m_A» ≈ auto = Automarke(kfz_nr typ baujahr)

Stellen/Prädikatssymbole:

s_1: Mietwilliger_kunde(kunde)
markenkapazität$_1$ = 10

s_2: Verfuegbares_auto(auto)
markenkapazität$_2$ = 20

s_3: Kunde_mit_vertrag(kunde auto)
markenkapazität$_3$ = 20

s_4: Verliehenes_auto(auto)
markenkapazität$_4$ = 20

Transitionen/Transaktionen:

t_1: vertragsabschluss

$<m_K>$ ≈ mietwilliger_kunde(kundenmarke(Name,adresse(Strasse,Wohnort),Alter))

$<m_A>$ ≈ verfügbares_auto(automarke(Kfz_nr,Typ,Baujahr))

Alter $\geq$ 21

$<m_K,m_A>$ ≈ kunde_mit_vertrag(kundenmarke(Name,adresse(Strasse,Wohnort),Alter), automarke(Kfz_nr,Typ,Baujahr))

$<m_A>$ ≈ verliehenes_auto(automarke(Kfz_nr,Typ,Baujahr))

<u>Fakten:</u>

$\langle m_{K.1} \rangle \approx$ $fakt_0$(1,mietwilliger_kunde(kundenmarke("Maier", adresse("Kaiserstr. 10","Karlsruhe"),40)))

$\langle m_{K.2} \rangle \approx$ $fakt_0$(1,mietwilliger_kunde(kundenmarke("Schmidt", adresse("Flughafenstr. 12","München"),19)))

$\langle m_{A.1} \rangle \approx$ $fakt_0$(1,verfügbares_auto(automarke("MA-K 460","Mazda 3f23",1986))

$\langle m_{A.2} \rangle \approx$ $fakt_0$(1,verfügbares_auto(automarke("MA-K 462","VW Golf",1987))

$\langle m_{A.3} \rangle \approx$ $fakt_0$(1,verliehenes_auto(automarke("MA-K 470","Ford Escort",1983))

$\langle m_{K.3}, m_{A.3} \rangle \approx$ $fakt_0$(1,kunde_mit_vertrag(kundenmarke("Schulte", adresse("Domstr. 4","Köln"),22),automarke("MA-K 470","Ford Escort",1983)))

Die Transition t_1 ist unter der Ausgangsmarkierung $M_0=FAK_0$ multipel aktiviert. Die Übergangsoperation ihrer zugeordneten Transaktion tr_1 kann mit genau zwei Variablenbelegungen vb_c aus der Menge $MOD_{1.0}=\{vb_c: c=1,2\}$ ausgeführt werden. Hiermit korrespondieren die beiden Schaltfarben "c=1" und "c=2" für die aktivierte Transition t_1. Die beiden Variablenbelegungen und ihre zugehörigen Schaltfarben bedeuten jeweils, daß mit dem mietwilligen Kunden "Maier" ein Mietvertrag abgeschlossen wird. Dabei wird ihm durch die erste Variablenbelegung der Mietwagen des Typs "Mazda 323", durch die zweite Variablenbelegung der Mietwagen des Typs "VW Golf" zur Verfügung gestellt. Obwohl auch der Kunde "Schmidt" einen Mietvertrag abschließen möchte, ist für ihn die Transition t_1 nicht aktiviert. Denn die Haupttestbedingung "Alter$\geq$21" wird von diesem erst neunzehnjährigen Kunden nicht erfüllt. Nachfolgend sind die beiden Variablenbelegungen vb_c angeführt, welche die zulässigen Schaltfarben für die Transition t_1 festlegen:

vb_1:	Alter	$\rightarrow$ 40
	Baujahr	$\rightarrow$ 1986
	Kfz_nr	$\rightarrow$ "MA-K 460"
	Name	$\rightarrow$ "Maier"
	Typ	$\rightarrow$ "Mazda 323"
	Straße	$\rightarrow$ "Kaiserstr. 10"
	Wohnort	$\rightarrow$ "Karlsruhe"
vb_2:	Alter	$\rightarrow$ 40
	Baujahr	$\rightarrow$ 1987
	Kfz_nr	$\rightarrow$ "MA-K 462"
	Name	$\rightarrow$ "Maier"
	Typ	$\rightarrow$ "VW Golf"
	Straße	$\rightarrow$ "Kaiserstr. 10"
	Wohnort	$\rightarrow$ "Karlsruhe"

Falls z.B. die Schaltfarbe "c=2" für die Transition t_1 mit der zugrundeliegenden Variablenbelegung vb_2 ausgewählt wird, erhält der Kunde "Maier" den Mietwagen des Typs "VW Golf". Für die Transformation der Ausgangsmarkierung M_0 in die Folgemarkierung M_1 gilt in transitions- und transaktionsbezogener Notation:

und:

$$M_0\,[t_1,2\rangle\,M_1$$

$$M_0\,[tr_1,vb_2\rangle\,M_1$$

Die Folgemarkierung M_1 ist identisch mit der Folgefaktenmenge FAK_1, die für das Beispielnetz nach dem Schalten der Transition t_1 gilt. Ausgangs- und Folgefaktenmenge unterscheiden sich durch die veränderten Stati des Kunden "Maier" und des Mietwagens vom Typ "VW Golf". Die Folgefaktenmenge $M_1 = FAK_1$ wird in der Sektion "Fakten" als nunmehr aktuelle Netzmarkierung präsentiert:

Fakten:

$<m_{K.2}> \approx$ $fakt_1$(1,mietwilliger_kunde(kundenmarke("Schmidt",
adresse("Flughafenstr. 12","München"),19)))

$<m_{A.1}> \approx$ $fakt_1$(1,verfügbares_auto(automarke("MA-K 460","Mazda 323",1986))

$<m_{A.2}> \approx$ $fakt_1$(1,verliehenes_auto(automarke("MA-K 462","VW Golf",1987))

$<m_{A.3}> \approx$ $fakt_1$(1,verliehenes_auto(automarke("MA-K 470","Ford Escort",1983))

$<m_{K.1},m_{A.2}> \approx$ $fakt_1$(1,kunde_mit_vertrag(kundenmarke("Maier",
adresse("Kaiserstr. 10","Karlsruhe"),40),
automarke("MA-K 462","VW Golf",1987)))

$<m_{K.3},m_{A.3}> \approx$ $fakt_1$(1,kunde_mit_vertrag(kundenmarke("Schulte",
adresse("Domstr. 4","Köln"),22),
automarke("MA-K 470","Ford Escort",1983)))

Abb. 52 auf der nächsten Seite zeigt die Netzgraphik für das Beispielnetz im Zustand "1" mit seiner Folgemarkierung $M_1 = FAK_1$.

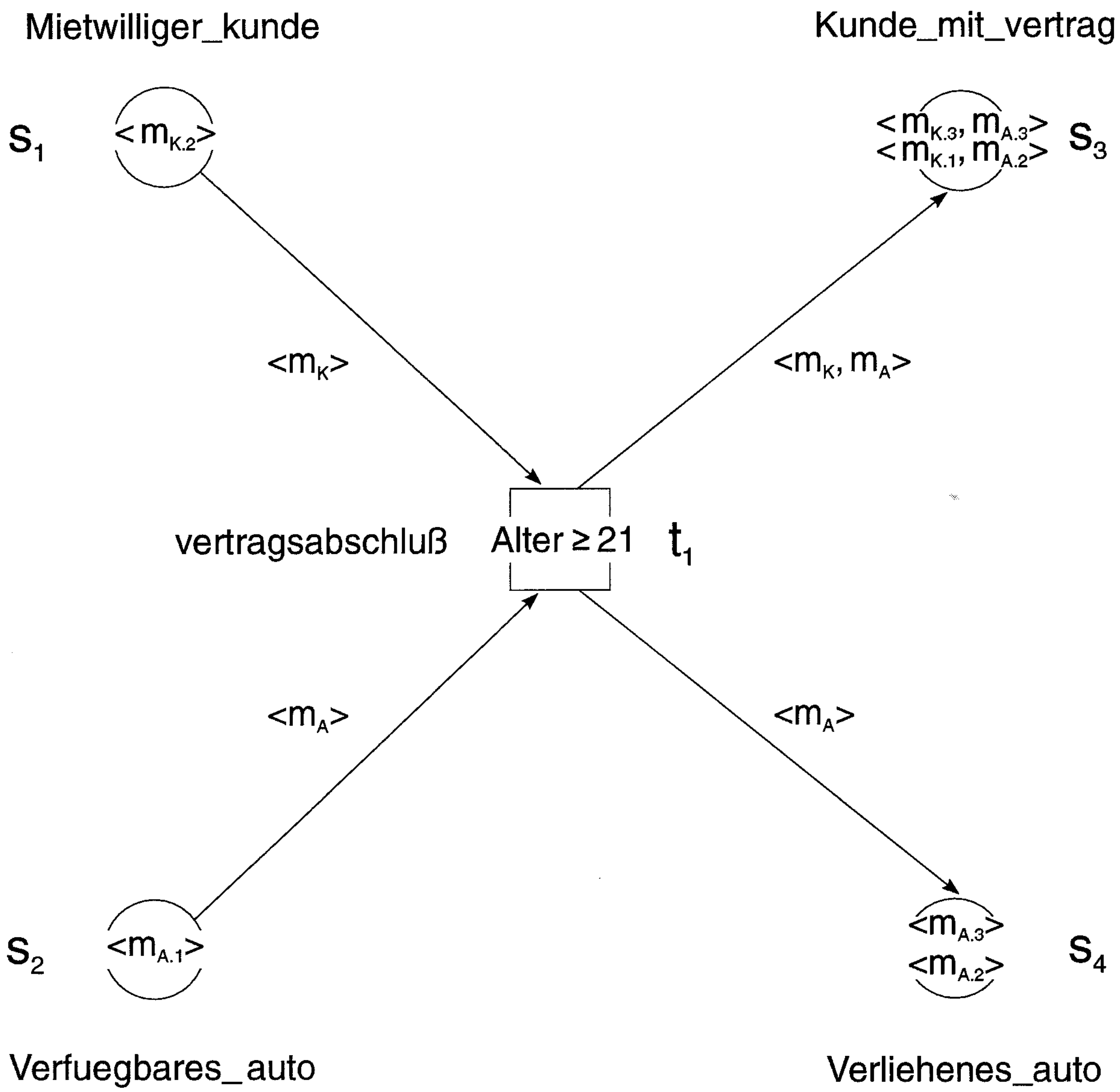

Abb. 52: Beispielnetz einer Autovermietung für den Folgezustand "r=1"

Abschließend wird die Kompaktheit und Transparenz der semi-graphischen Netzdarstellung verdeutlicht. Zu diesem Zweck wird dasselbe Beispiel der Autovermietung so formuliert, daß es unmittelbar eine Ausprägung des algebraisch-prädikatenlogischen Definitionsschemas für Synthetische Netze darstellt. Obwohl dabei schon auf früher zugelassene Vereinfachungen für Definitionen konkreter Synthetischer Netze zurückgegriffen wird[54)], fällt die Definition des Beispielnetzes immer noch recht umfangreich aus:

Beispielnetz $SN=(TOP,SPEC_{MSIG};BES,M_0)$ mit:

1) Netztopologie TOP=(S,T;F):

$S = \{s_1,s_2,s_3,s_4\}$

$T = \{t_1\}$

$F = \{(s_1,t_1),(s_2,t_1),(t_1,s_3),(t_1,s_4)\}$

2) Netzspezifikation $SPEC_{MSIG}=(SO,OP,PRÄ;OBF,OPF,FAK_0;RES;TR)$:

<u>sorts</u>: adressfeld
alter
auto
baujahr
kfz_nr
kunde
name
strasse
typ
wohnort

SO = {adressfeld,alter,auto,baujahr,kfz_nr,kunde,name,strasse,typ,wohnort}

<u>Ops</u>: Adresse: strasse wohnort → adressfeld

Kundenmarke: name adressfeld alter → kunde

Automarke: kfz_nr typ baujahr → auto

OP = {Adresse,Kundenmarke,Automarke}

<u>Präs</u>: Mietwilliger_kunde(kunde)

Verfuegbares_auto(auto)

Kunde_mit_vertrag(kunde auto)

Verliehenes_auto(auto)

PRÄ = {Mietwilliger_kunde,Verfuegbares_auto,Kunde_mit_vertrag,Verliehenes_auto}

<u>OBs</u>:

$OB_{adressfeld}$ = SYMBOL
OB_{alter} = INTEGER
OB_{auto} = SYMBOL
$OB_{baujahr}$ = INTEGER
OB_{kfz_nr} = STRING
OB_{kunde} = SYMBOL
OB_{name} = STRING
$OB_{strasse}$ = STRING
OB_{typ} = STRING
$OB_{wohnort}$ = STRING

$OBF = (OB_{adressfeld}, OB_{alter}, OB_{auto}, OB_{baujahr}, OB_{kfz\_nr}, OB_{kunde}, OB_{name}, OB_{strasse}, OB_{typ}, OB_{wohnort})$

<u>ops</u>:

adresse: $OB_{strasse} \text{ x } OB_{wohnort} \rightarrow OB_{adressfeld}$
$(ob_1, ob_2) \rightarrow adresse(ob_1, ob_2) = ob_3$

kundenmarke: $OB_{name} \text{ x } OB_{adressfeld} \text{ x } OB_{alter} \rightarrow OB_{kunde}$
$(ob_1, ob_2, ob_3) \rightarrow kundenmarke(ob_1, ob_2, ob_3) = ob_4$

automarke: $OB_{kfz\_nr} \text{ x } OB_{typ} \text{ x } OB_{baujahr} \rightarrow OB_{auto}$
$(ob_1, ob_2, ob_3) \rightarrow automarke(ob_1, ob_2, ob_3) = ob_4$

OPF = (adresse,kundenmarke,automarke)

<u>präs</u>:

mietwilliger_kunde: OB_{kunde}

fakt$_0$(1,mietwilliger_kunde(kundenmarke("Maier",
adresse("Kaiserstr. 10","Karlsruhe"),40)))

fakt$_0$(1,mietwilliger_kunde(kundenmarke("Schmidt",
adresse("Flughafenstr. 12","München"),19)))

FAK$_{mietwilliger_kunde.0}$ = ...
{fakt$_0$(1,mietwilliger_kunde(kundenmarke("Maier",
adresse("Kaiserstr. 10","Karlsruhe"),40))),
fakt$_0$(1,mietwilliger_kunde(kundenmarke("Schmidt",
adresse("Flughafenstr.12","München"),19)))}

verfuegbares_auto: OB_{auto}

fakt$_0$(1,verfuegbares_auto(automarke("MA-K 460","Mazda 323",1986))

fakt$_0$(1,verfuegbares_auto(automarke("MA-K 462","VW Golf",1987))

FAK$_{verfuegbares_auto.0}$ = ...
{fakt$_0$(1,verfuegbares_auto(automarke("MA-K 460","Mazda 323",1986)),
fakt$_0$(1,verfuegbares_auto(automarke("MA-K 462","VW Golf",1987)}

kunde_mit_vertrag: OB_{kunde} x OB_{auto}

$fakt_0$(1,kunde_mit_vertrag(kundenmarke("Schulte",
adresse("Domstr. 4","Köln"),22),automarke("MA-K 470","Ford Escort",1983)))

$FAK_{kunde\_mit\_vertrag.0}$ = ...
{$fakt_0$(1,kunde_mit_vertrag(kundenmarke("Schulte",
adresse("Domstr. 4","Köln"),22),
automarke("MA-K 470","Ford Escort",1983)))}

verliehenes_auto: OB_{auto}

$fakt_0$(1,verliehenes_auto(automarke("MA-K 470","Ford Escort",1983))

$FAK_{verliehenes\_auto.0}$ = ...
{$fakt_0$(1,verliehenes_auto(automarke("MA-K 470","Ford Escort",1983))}

FAK_0 = ($FAK_{mietwilliger\_kunde.0}$,$FAK_{verfuegbares\_auto.0}$,$FAK_{kunde\_mit\_vertrag.0}$,$FAK_{verliehenes\_auto.0}$)

res: RES = $FOR_{standard}$

trans: vertragsabschluss = ...

(VB(vertragsabschluss) = {Mietwilliger_kunde,Verfuegbares_auto},
$MTAV_{vertragsabschluss}$ = ...
{$MTAV_{Mietwilliger\_kunde.vertragsabschluss}$ = ...
{(1,mietwilliger_kunde(kundenmarke(Name,
adresse(Strasse,Wohnort),Alter)))},
$MTAV_{Verfuegbares\_auto.vertragsabschluss}$ = ...
{(1,verfuegbares_auto(automarke(Kfz_nr,Typ,Baujahr)))}},
IB(vertragsabschluss) = $\emptyset$,
$MTAI_{vertragsabschluss}$ = $\emptyset$,
NB(vertragsabschluss) = {Kunde_mit_vertrag,Verliehenes_auto},
$MTAN_{vertragsabschluss}$ = ...
{$MTAN_{Kunde\_mit\_vertrag.vertragsabschluss}$ = ...
{(1,kunde_mit_vertrag(kundenmarke(Name,
adresse(Strasse,Wohnort),Alter),
automarke(Kfz_nr,Typ,Baujahr)))},
$MTAN_{Verliehenes\_auto.vertragsabschluss}$ = ...
{(1,verliehenes_auto(automarke(Kfz_nr,Typ,Baujahr)))}},
$RES_{vertragsabschluss}$ = {Alter$\geq$21})

$TR = \{vertragsabschluss\}$

mit:

$MTA = (MTAV_{Mietwilliger\_kunde.vertragsabschluss}, MTAV_{Verfuegbares\_auto.vertragsabschluss},$
$MTAN_{Kunde\_mit\_vertrag.vertragsabschluss}, MTAN_{Verliehenes\_auto.vertragsabschluss})$

3) Netzbeschriftung BES=(bsp,bsk,btt,bfm):

bsp: $S \rightarrow PRÄ$
$s_1 \rightarrow bsp(s_1) = Mietwilliger\_kunde$
$s_2 \rightarrow bsp(s_2) = Verfuegbares\_auto$
$s_3 \rightarrow bsp(s_3) = Kunde\_mit\_vertrag$
$s_4 \rightarrow bsp(s_4) = Verliehenes\_auto$

bsk: $S \rightarrow \mathcal{N}_+$
$s_1 \rightarrow bsk(s_1) = 10$
$s_2 \rightarrow bsk(s_2) = 20$
$s_3 \rightarrow bsk(s_3) = 20$
$s_4 \rightarrow bsk(s_4) = 20$

btt: $T \rightarrow TR$
$t_1 \rightarrow vertragsabschluss$

bfm: $F \rightarrow MTA$
$(s_1,t_1) \rightarrow bfm(s_1,t_1) = MTAV_{Mietwilliger\_kunde.vertragsabschluss}$
$(s_2,t_1) \rightarrow bfm(s_2,t_1) = MTAV_{Verfuegbares\_auto.vertragsabschluss}$
$(t_1,s_3) \rightarrow bfm(t_1,s_3) = MTAV_{Kunde\_mit\_vertrag.vertragsabschluss}$
$(t_1,s_4) \rightarrow bfm(t_1,s_4) = MTAV_{Verliehenes\_auto.vertragsabschluss}$

4) Ausgangsmarkierung M_0:

M_0: $S \rightarrow FAK_0$
$s_1 \rightarrow M_0(s_1) = FAK_{mietwilliger\_kunde.0}$
$s_2 \rightarrow M_0(s_2) = FAK_{verfuegbares\_auto.0}$
$s_3 \rightarrow M_0(s_3) = FAK_{kunde\_mit\_vertrag.0}$
$s_4 \rightarrow M_0(s_4) = FAK_{verliehenes\_auto.0}$

Anmerkungen zum Kapitel:

1) Vgl. die Anmerkungen zum Postulat der vollständigen Explizierung von konzeptrelevanten Entitäten. Vgl. auch die korrespondierende Kritik an mangelhafter Konzeptexplizierung bei konventionellen Netzdefinitionen, die sich vor allem auf Integritätsbedingungen und Schaltregeln bezieht.

2) Diese Redundanzen waren im Definitionsschema keineswegs überflüssig, sondern ermöglichten erst die klare Strukturierung des Definitionsschemas. Damit entsprechen sie dem Prinzip kontrollierter Redundanz, auf das bereits hingewiesen wurde. Beispielsweise sind die Einteilungen der Nachbarschaften von Transitionen in Eingangs-, Informations- und Ausgangsstellen sowie die Einteilungen der Operationsbereiche von Transaktionen in Prädikatssymbole ihrer Eingangs-, Informations- bzw. Ausgangsbereiche redundant, weil die Stellen und Prädikatssymbole durch die Integritätsbedingungen für Synthetische Netze einander eineindeutig zugeordnet werden. Trotzdem wurde im Definitionsschema diese Redundanz nicht beseitigt, um die Netztopologie mit ihren Transitionen und die Netzspezifikation mit ihren Transaktionen als fundamentale Netzkomponenten jeweils vollständig und voneinander unabhängig einführen zu können.

3) Dazu gehören vor allem die Abstraktion von allen Variablenbelegungen vb_c in den Restriktionsformeln der Transaktionen und die Vernachlässigung des allgemeinen Übergangsschemas. Beide werden im Programmpaket PASIPP auf der Grundlage des PROLOG-spezifischen Unifizierungs- und Resolutionskonzepts realisiert, ohne explizit aufgeführt werden zu müssen.

4) Vgl. dazu die Erläuterung, daß die Menge IB der netzkonstitutiven Integritätsbedingungen für alle Synthetische Netze in derselben Weise gilt und daher nicht explizit notiert zu werden braucht.

5) Dies bietet sich vor allem dann an, wenn der formalsprachliche Transaktionsname "tr_v" durch einen natürlichsprachlichen Transaktionsnamen ersetzt wird.

6) Für Kopien der Basismarke ist kein differenzierender Index erforderlich, weil ihre strukturlosen Kopien notwendig identisch sind. Dennoch kann ihnen derselbe differenzierende Index d=0 zugeordnet werden. Dies ist unter dem technischen Aspekt hilfreich, daß dann in den nachfolgend präsentierten Formeln nicht zwischen Markenkopien unterschieden werden muß, die entweder einen oder aber keinen differenzierenden Index besitzen. Daher lassen sich alle Kopien der Basismarke ebenso durch den Ausdruck "$m_{0.0}$" darstellen.

7) Die Indexgeltung für das *Gesamt*netz bedeutet, daß zwei Kopien derselben Marke, die innerhalb desselben Netzes denselben Index "d" erhalten haben, auf jeden Fall identische Markenkopien darstellen müssen. Die Umkehrung gilt zwar in der Regel, aber nicht auf jeden Fall. Auf einen Ausnahmefall, in dem identische Kopien derselben Marke dennoch verschiedene Indizes "d" erhalten, wird in einer anderen Anmerkung hingewiesen.

8) Der Prädikatssymbolname ist wegen der bijektiven Zuordnung von Stellen und Prädikatssymbolen durch die jeweils betrachtete Stelle eindeutig festgelegt.

9) Dies bedeutet insbesondere, daß die Attributausprägungen der Kopien von Attributmarken in der Netzgraphik nicht expliziert werden. Da alle strukturierten Marken letztlich aus Attributmarken bestehen, werden strukturierte Marken in der Netzgraphik niemals vollständig expliziert.

10) Diese Kopien sind von vornherein "vollständig expliziert", da überhaupt keine explizierbare innere Markenstruktur definiert ist.

11) Der Index "0" wird für die Kopien der Basismarke reserviert.

12) Neutrale Summanden mit der Multiplizität $mu_{u.n.d}=0$ werden vernachlässigt. Einheitssummanden mit der Multiplizität $mu_{u.n.d}=1$ werden ohne diese Multiplizität notiert. Vgl. dazu die Ausführungen zur Notation von Multimengen. Die Verwendung formaler Summen entspricht genau der vorherrschenden Notationsweise von Prädikat/Transition-Netzen. Daher trägt ihre Benutzung in der graphischen Darstellung von Synthetischen Netzen dazu bei, beide Netzklassen in notationaler Hinsicht eng aufeinander zu beziehen.

13) Dies bedeutet keinen Informationsverlust. Denn das betroffene Prädikatssymbol $Prä_u$ steht für jedes Gewicht $MTAV_{u.v}$ oder $MTAI_{u.v}$ einer Kante (s_m,t_n) und für jedes Gewicht $MTAN_{u.v}$ einer Kante (t_n,s_m) als dasjenige Prädikatssymbol fest, das in der Netzbeschriftung der Stelle s_m zugeordnet ist.

14) Allerdings müssen variable Kopien derselben Marke weiterhin durch einen differenzierenden Index "d" unterschieden werden, wenn sie sich dadurch voneinander abheben, daß sie in ihrer inneren, nicht explizierten Struktur *verschiedene* Variablen enthalten. In diesem Fall drückt der differenzierende Index aus, daß die variablen Markenkopien unter derselben Variablenbelegung durch verschiedene konstante Markenkopien gebunden werden dürfen. Dies trifft insbesondere dann zu, wenn eine variable Markenkopie $m_{s(u.k).d}$ in einem Kantengewicht mehrfach - und zwar in verschiedenen Varianten - vorkommt. Dies kann immer dann der Fall sein, wenn die formale Summe des Kantengewichts aus zwei oder mehr Summanden besteht.

Wegen der Lokalität der Variablendefinition in Synthetischen Netzen kann jedoch auf den differenzierenden Index "d" für variable Kopien derselben Marke verzichtet werden, *obwohl* sie verschiedene Variablen enthalten, *falls* die variablen Markengewichte nicht in den Kantengewichten derselben Transition vorkommen. Denn unterschiedliche Vorkommnisse derselben Variablen dürfen aufgrund der lokalen Definitionscharakteristik durch verschiedene Konstanten belegt werden, sobald die Variablenvorkommnisse nicht zur lokalen Vorkommnisgruppe derselben Transition gehören. Dabei umfaßt die Vorkommnisgruppe einer Transition alle Variablenvorkommnisse in der Deklaration der transitionszugehörigen Transaktion. Dazu gehören insbesondere die Variablenvorkommnisse in den Gewichten derjenigen Kanten, welche die Transition mit ihren benachbarten Stellen verknüpfen, und in den Restriktionsformeln, die das Schalten der Transition einschränken.

15) Zumeist gehören zur Deklaration der transitionszugehörigen Transaktion mehr Restriktionsformeln, als sich in einer Netzgraphik auf einer Seite übersichtlich unterbringen lassen. Dies gilt selbst dann, wenn nur die Restriktionsformeln von Bestimmungsgleichungen und Haupttestbedingungen berücksichtigt werden. Daher werden zumeist nur diejenigen Restriktionsformeln in der Netzgraphik ausgewiesen, die für das Verständnis des Schaltverhaltens der jeweils betroffenen Transition besonders wichtig erscheinen. Alle anderen Restriktionsformeln (aus Bestimmungsgleichungen und Haupttestbedingungen) können dagegen der Netzlegende entnommen werden. Darüber hinaus läßt sich in der Netzgraphik ein Bezug zu den nicht ausgewiesenen Formeln dadurch herstellen, daß in transitionsartigen Knoten der Netzgraphik Aggregatformeln "for_<Formelkennzeichnung>" notiert werden. In der Netzlegende werden diesen Aggregatformeln die jeweils zugehörigen, in der Netzgraphik aber nicht enthaltenen Teilformeln durch definitorische Äquivalenzen zugeordnet. Diese Zuordnungstechnik wird in der später präsentierten Fallstudie häufige Verwendung finden.

16) Alle anderen Restriktionsformeln werden weder in der Netzgraphik noch in der Netzlegende aufgeführt, weil sie keinen besonderen Informationswert besitzen. Vielmehr gelten sie als implizit vereinbart durch jene einheitlichen Voreinstellungen, die im voranstehenden Kapitel für das Kernkonzept Synthetischer Netze eingeführt wurden. Sie bestehen aus der Formulierungen von Testbedingungen durch immer gültige Tautologien oder aus Überprüfungen der Einhaltung von Markenkapazitäten.

17) Eine solche optische Separation der Restriktionsformeln von Haupttestbedingungen einerseits und Bestimmungsgleichungen andererseits ist jedoch nicht notwendig. Denn die Notation von Restriktionsforneln legt deren Bedingungs- oder Gleichungsqualität bereits eindeutig fest: Alle Restriktionsformeln, die keine Gleichungen darstellen, können ohnehin nur Haupttestbedingungen darstellen. Gleichungen werden dagegen entweder mit dem Operator "=" oder aber mit dem Operator ":=" notiert, und zwar je nachdem, ob es sich um eine Haupttestbedingung bzw. Bestimmungsgleichung handelt.

18) Darüber hinaus erfüllt der differenzierende Index "d" der Markenkopien $m_{s(u.k).d}$, die mit $k \in \}1,...,K_u\}$ im Argument $(m_{s(u.1).d},...,m_{s(u.Ku).d})$ einer atomaren Formel $prä_u$ vorkommen, noch eine besondere Funktion. Dies gilt allerdings nur für eine solche atomare Formel $prä_u(m_{s(u.1).d},...,m_{s(u.Ku).d})$, die zugleich in *mehreren* Kantengewichten enthalten ist. Hierfür gilt eine besondere Vereinbarung. Im Regelfall wird der differenzierende Index "d" nur angewendet, um verschiedene Kopien derselben Marke voneinander zu unterscheiden: Wenn zwei Kopien derselben Marke denselben Index "d" besitzen, dann stellen sie notwendig identische Markenkopien dar. Es wurde jedoch schon erwähnt, daß die Umkehrung nicht immer zutreffen muß. Statt dessen existiert ein Sonderfall, in dem zwei identische Kopien derselben Marke dennoch unterschiedliche differenzierende Indizes "d" erhalten. In diesem Sonderfall leistet der Index "d" nicht die Differenzierung zwischen verschiedenen Markenkopien. Vielmehr differenziert er nunmehr *unterschiedliche Multiplizitäten* derselben atomaren Formel $prä_u$, die in mehreren Kantengewichten enthalten ist.
Eine solche Differenzierung wird erforderlich, wenn eine 1-Schleife besonderer Art vorliegt. Ausgangspunkt ist - wie für jede 1-Schleife - eine Stelle s_m, die sowohl zum Vorbereich VB_n als auch zum Nachbereich NB_n einer Transition t_n gehört. Der Stelle s_m ist das K_u-stellige Prädikatssymbol $Prä_u$ zugeordnet. Die beiden, entgegengesetzt gerichteten Kanten zwischen der Stelle s_m und der Transition t_n sind mit jeweils einer Multimenge aus atomaren Formeln derart gewichtet, daß gilt: Beide Multimengen enthalten mindestens eine atomare Formel $prä_u(m_{s(u.1)},...,m_{s(u.Ku)})$ gemeinsam, jedoch mit unterschiedlicher Multiplizität $mu_{u.n}$. Diese *unterschiedlichen* Multiplizitäten erfordern den differenzierenden Index "d". Denn es ist formal unzulässig, die Multiplizitäten derselben atomaren Formel $prä_u(m_{s(u.1)},...,m_{s(u.Ku)})$ in den Gewichten der beiden verschiedenen Kanten der o.a. 1-Schleife durch *denselben* Ausdruck "$mu_{u.n}$" zu notieren, *obwohl* es sich per constructionem um *unterschiedliche* Multiplizitäten handelt. Die beiden Teilindizes "u" und "n" für das Prädikatssymbol $Prä_u$ der Stelle s_m bzw. für die Transition t_n erlauben die erforderliche Ausdrucksdifferenzierung nicht, weil sie in jeder 1-Schleife notwendig identisch sind. Deshalb muß der differenzierende Index "d" hinzugenommen werden, um die unterschiedlichen Multiplizitäten der beiden Kanten durch die Ausdrücke $mu_{u.n.d}$ formal korrekt auseinanderzuhalten. Z.B. lassen sich die Indizes durch d=1 und d=2 für die Multiplizitäten der Ein- bzw. der Ausgangskante von Transition t_n festlegen. Dann umfassen die Gewichte dieser beiden Kanten - in der Notation als formale Summen - die Summanden $mu_{u.n.1} \bullet prä_u(m_{s(u.1).1},..., m_{s(u.Ku).1})$ bzw. $mu_{u.n.2} \bullet prä_u(m_{s(u.1).2},...,m_{s(u.Ku).2})$. Laut Voraussetzung gilt zwar $prä_u(m_{s(u.1).1},...,m_{s(u.Ku).1}) = prä_u(m_{s(u.1).2},..., m_{s(u.Ku).2})$, also auch $m_{s(u.k).1} = m_{s(u.k).2}$ für alle $k \in \{1,...,K_u\}$. Es liegen also identische Markenkopien vor. Jedoch wird

der differenzierende dritte Index dadurch gerechtfertigt, daß $mu_{u.n.1} \neq mu_{u.n.2}$ zutrifft. Diese formal korrekte Ungleichung tritt an die Stelle der formal unzulässigen Ungleichung $mu_{u.n} \neq mu_{u.n}$.

19) Beispielsweise besitzen in dem unten exemplarisch angeführten Synthetischen Netz die sechs Markenkopien $m_{A.1}$, $m_{A.2}$, $m_{A.3}$, $m_{K.1}$, $m_{K.2}$ und $m_{K.3}$ eindeutig differenzierende Gesamtindizes. Dazu reichen die differenzierenden Indizes "d" mit $d \in \{1,2,3\}$ für die beiden involvierten Markenarten "A" und "K" aus. Die beiden Indexmengen $D_A = \{d: d=1,2,3\}$ und $D_K = \{d: d=1,2,3\}$ erfüllen also die geforderte differenzierende Funktion, ohne disjunkt zu sein.

20) Allerdings darf der Index "d" der Markenkopie $m_{s.d}$ nur vernachlässigt werden, falls sich im gesamten Netz unter der Ausgangsmarkierung $M_0 = FAK_0$ von der Markenart "s" keine *verschiedenen* Kopien befinden. Für die Markenart "s" können dann zwar noch mehrere Markenkopien existieren; diese müssen dann aber alle identische sein.

21) Vgl. auch die nachfolgende exemplarische Erläuterung der vollständigen Explizierung bei Kantengewichten.

22) Dies entspricht der früher erhobenen Forderung nach möglichst weitgehender Explizierung aller modellierungsrelevanten Sachverhalte. Allerdings wurde dort auch der Vorbehalt geäußert, zugunsten einer kontrollierten Explizitheit Teile des modellierungsrelevanten Wissens implizit vorzuhalten, das sich im Bedarfsfall durch Inferenzen erschließen läßt. Dieser Aspekt wird hier in zweifacher Hinsicht gewürdigt. Einerseits wird nachfolgend die anonyme Variable zugelassen, um trotz vollständiger Termexplizierung solche Modellierungsaspekte vernachlässigen zu können, die im aktuellen Argumentationskontext nicht relevant sind. Andererseits wird der Kompaktheitsvorteil unvollständiger Explizierung in der anschließenden Vereinbarung für die Darstellung von Prädikatssymbolen genutzt. Insofern wird ein Kompromiß zwischen vollständiger Explizierung der Argumente von Kantengewichten und Faktenmengen sowie unvollständiger Explizierung der Argumente von Prädikatssymbolen getroffen: Die spezielle Darstellungsweise Synthetischer Netze verwirklicht das Prinzip kontrollierter Explizitheit.
Allerdings räumt der Verf. ein, daß es sich bei der Festlegung des Explizierungsausmaßes für Kantengewichte, Faktenmengen und Prädikatssymbole letztlich um eine willkürliche Basisentscheidung handelt. Diese Entscheidung hätte auch in anderer Weise getroffen werden können. Beispielsweise hätte die vollständige Explizierung der Prädikatssymbole in der Netzgraphik gefordert werden können. Die Abwägung zugunsten der oben getroffenen Festlegung resultiert aus den praktischen Erfahrungen des Verf. hinsichtlich der Kompaktheit und Klarheit von alternativen - aber äquivalenten - Netzformulierungen. Diese Abwägung bleibt aber letztlich ein stark subjektiv gefärbtes Erfahrungsurteil, das hier nicht durch weiterführende Begründung "objektiviert" wird. Ein solcher Objektivierungsversuch liefe auf den Anspruch heraus, für die Netzdarstellung ein optimales Explizierungsniveau *begründen* zu können. Ein solcher Anspruch wurde bereits - unter dem umfassenderen Aspekt eines optimalen Komplexionsniveaus - thematisiert und als nicht einlösbar verworfen.

23) Alle Informationen über die modellierten realen Objekte werden aufgrund der früher entfalteten Marken-Ontologie in den Marken eines Synthetischen Netzes und deren Kopien konserviert. Durch die Verwendung der anonymen Variable wird keine Information, die in der Deklaration einer Markenkopie enthalten ist, zerstört. Diese Information besteht in der Marke fort. Nur wird sie beim Schalten einer Transition mit einem Kantengewicht, das die anonyme Variable enthält, nicht zur Kenntnis genommen.

24) Beispielsweise sind die Objektmengen aller Attribut- und Kompositmarken als $OB_{att\_marke} = SYMBOL$ bzw. $OB_{str\_marke} = SYMBOL$ festgelegt. Daher werden diese Objektmengenfestlegungen in der Legende nicht explizit aufgeführt. Die Operationssymbole identischer Abbildungen, die Attributmarken als strukturierte Marken ausweisen sowie die Basismarke und alle strukturierten Marken als sortierte Marken klassifizieren, gelten ebenso als implizit vereinbart. Auch die Objektmenge der Basismarke könnte mit $OB_{bas\_marke} = SYMBOL$ als implizit vereinbart unterstellt werden. Dennoch wird sie hier weiterhin aus zwei Gründen explizit in der Netzlegende angeführt. Erstens läßt sie den expliziten Teil der allgemeinen Netzlegenden-Definition geschlossener erscheinen. Zweitens erlaubt sie den unmittelbaren Anschluß späterer Erörterungen, die sich mit der Implementierung von Synthetischen Netzen befassen. Vgl. dazu insbesondere die Anmerkung zur Darstellung der Basismarke in der Programmiersprache Turbo-PROLOG.

25) Vgl. die Anmerkung, die sich mit den allgemeinen Objektmengen aus der Programmiersprache Turbo-PROLOG befaßte. Hinsichtlich der Objektmenge REAL gilt es, einige Besonderheiten zu beachten. Erstens umgreift sie keineswegs alle Realzahlen. Darauf wurde schon an früherer Stelle hingewiesen. Aber sie gestattet immerhin, Dezimalzahlen innerhalb eines vorgegebenen Intervalls darzustellen. Die Intervallgrenzen hängen davon ab, auf welcher Soft- und Hardwarebasis Netzmodelle implementiert werden sollen. Es wird unterstellt, daß sich diese Intervallgrenzen auf die Modellgestaltung nicht restriktiv auswirken. Darüber hinaus werden alle Dezimalzahlen kohärent zur Notationsweise in der Programmiersprache Turbo-PROLOG dargestellt: Die Ganzzahlen und die Dezimalstellen werden in jeder Dezimalzahl durch einen Punkt (".") voneinander getrennt. Diese Notation ist erforderlich, weil das Komma (",") bereits für die Separation von Attributausprägungen und Markenkopien in k-Tupeln mit $k \geq 2$ vergeben ist. Eine Dezimalzahl, die eine einzelne Attributausprägung darstellt, darf daher nicht selbst mit der Hilfe eines Kommas ausgedrückt werden. Vgl. zur Notation von Dezimalzahlen ("Realzahlen") in der Programmiersprache Turbo-PROLOG z.B. KINNEBROCK (1988), S. 121.

26) Implizit wird für alle derivativ definierten Attributsorten "attribut_q" der globale Definitionsbereich $OB_{att.q}$=SYMBOL unterstellt. Seine explizite Angabe ist nicht erforderlich, weil die zulässigen Attributausprägungen induktiv zusammengesetzt werden können aus den Ausprägungen jener Attributsorten, die an der derivativen Definition der jeweils betrachteten Attributsorte beteiligt sind.

27) Falls z.B. die Markensorte att_marke_a einer Attributmarke sowohl als Zielsorte des Operationssymbols "Marke_3" als auch als Zielsorte des Operationssymbols "Marke_7" definiert ist, so wird dies notiert als:

$$\begin{aligned}\text{att\_marke}_a = \; & \text{Marke}_3(\text{attribut}_{q(3.1)} \ldots \text{attribut}_{q(3.K3)}); \\ & \text{Marke}_7(\text{attribut}_{q(7.1)} \ldots \text{attribut}_{q(7.K7)})\end{aligned}$$

Diese Notationsweise lehnt sich an die multiple Objektdeklaration in der Programmiersprache Turbo-PROLOG an; vgl. KINNEBROCK (1988), S. 44f.

28) Es gilt in dieser Arbeit grundsätzlich die Vereinbarung, daß Ausdrücke, die durch die Klammern "[" und "]" eingeschlossen werden, optionale Ausdrücke darstellen, die benutzt werden können aber nicht müssen. Den beiden Klammern dürfen in speziellen Kontexten allerdings auch abweichende Bedeutungen zugeordnet werden; vgl. z.B. ihre Verwendung zur Listennotation von Multimengen.

29) Ebenso kann den Kopien der Basismarke durch die Notation "«∅»" die Darstellung "∅" in der Netzgraphik zuordnen.

30) Für die eineindeutige Zuordnung werden folgende definitorischen Zuordnungen vereinbart:

a) für die Basismarke:

$$«m_0» \approx \text{bas\_marke: SYMBOL}$$

$$\begin{aligned}:\Leftrightarrow \quad & \text{bas\_marke}, \; OB_{\text{bas\_marke}} = \text{SYMBOL} \\ & \text{marke}_0\text{:} \; \rightarrow OB_{\text{bas\_marke}}, \; \text{marke}_0() = m_0\end{aligned}$$

$$«\varnothing» \approx \text{bas\_marke: SYMBOL}$$

$$\begin{aligned}:\Leftrightarrow \quad & \text{bas\_marke}, \; OB_{\text{bas\_marke}} = \text{SYMBOL} \\ & \text{marke}_0\text{:} \; \rightarrow OB_{\text{bas\_marke}}, \; \text{marke}_0() = \varnothing\end{aligned}$$

b) für die Attributmarken mit a∈ {1,..,A}:

$$«m_a» \approx \text{att\_marke}_a = \text{Marke}_{j(a)}(\text{attribut}_{q(j(a).1)}\text{attribut}_{q(j(a).Kj(1))}) \; [;\ldots]$$

$$\begin{aligned}:\Leftrightarrow \quad & \text{att\_marke}_a, \quad OB_{\text{att\_marke}} = \text{SYMBOL} \\ & \text{Marke}_{j(a)}\text{:} \quad \text{attribut}_{q(j(a).1)} \ldots \text{attribut}_{q(j(a).Kj(a))}) \rightarrow \text{att\_marke}_a \\ & \text{marke}_{j(a)}\text{:} \quad OB_{att.q(j(a).1)} \times \ldots \times OB_{att.q(j(a).Kj(a))} \rightarrow OB_{\text{att\_marke}} \\ & \qquad (at_{a.1},\ldots,at_{a.Ka}) \rightarrow \text{marke}_{j(a)}(at_{a.1},\ldots,at_{a.Ka}) = m_a \; [;\ldots]\end{aligned}$$

c) für die Kompositmarken mit b∈ {1,..,B}:

$$«m_{A+b}» \approx \text{str\_marke}_{A+b} = \text{Struk}_{j(A+b)}(\text{str\_marke}_{s(j(A+b).1)} \ldots \text{str\_marke}_{s(j(A+b).Kj(A+b))}) \; [;\ldots]$$

$$\begin{aligned}:\Leftrightarrow \quad & \text{str\_marke}_{A+b} \; OB_{\text{str\_marke}} = \text{SYMBOL} \\ & \text{Struk}_{j(A+b)}\text{:} \quad \text{str\_marke}_{s(j(A+b).1)} \ldots \text{str\_marke}_{s(j(A+b).Kj(A+b))}) \rightarrow \text{str\_marke}_{A+b} \\ & \text{struk}_{j(A+b)}\text{:} \quad OB_{\text{str\_marke}} \times \ldots \times OB_{\text{str\_marke}} \rightarrow OB_{\text{str\_marke}} \\ & \qquad (m_{s(A+b.1)},\ldots,m_{s(A+b.KA+b)}) \rightarrow \text{struk}_{j(A+b)}(m_{s(A+b.1)},\ldots,m_{s(A+b.KA+b)}) = m_{A+b} \; [;\ldots]\end{aligned}$$

Im Fall von Attribut- und Kompositmarken wird von den differenzierenden Indizes "d", die unterschiedliche Kopien derselben strukturierten Marke unterscheiden können, abgesehen. Denn es werden hier jeweils *alle* Kopien derselben strukturierten Marke angesprochen. Dabei interessieren die individuellen Ausprägungen der Markenkopien nicht.

31) Auf die vollständige Explizierung weist das Superskript "ex" hin.
Wenn es sich beispielsweise bei der Markenkopie $m_{s(u.k).d}{}^{ex}$ um die Kopie einer Attributmarke "att_marke$_{s(u.k).d}$" aus originär definierten Attributsorten handelt und wenn diese Markenkopie in einem atomaren Formelvorkommnis $prä_u(...,m_{s(u.k).d}{}^{ex},...)$ enthalten ist, dann wird die Markenkopie als vollständig explizierter Attributausprägungsvektor mit Erzeugungsoperator $marke_{j(s(u.k))}(...)$ notiert: $marke_{j(s(u.k))}(at_{j(s(u.k)).1.d},...,at_{j(s(u.k)).Kj(s(u.k)).d})$. Diese vollständig explizierte Darstellungsweise wird in der Sektion "Transitionen/Transaktionen" durch die Ergänzung "$\approx ... m_{s(u.k).d} ...$" mit ihrer impliziten Repräsentationsweise $m_{s(u.k).d}$ in der Netzgraphik eineindeutig verknüpft.
Falls eine Komponente aus einer vollständig explizierten Markenkopie nicht von Interesse ist, kann sie durch die anonyme Variable "_" vertreten werden. Auf diese Weise bleibt die vollständige Explizierung erhalten, ohne die irrelevanten Komponenten näher betrachten zu müssen.

32) Prima facie könnte diese Zuordnung redundant erscheinen. Dieser Eindruck täuscht jedoch. Denn in den Ausdrücken $prä_u(m_{s(u.1).d}{}^{ex},...,m_{s(u.Ku).d}{}^{ex})$ der Kantengewichte aus der Netzlegende werden die Terme $m_{s(u.k).d}{}^{ex}$ für alle $k \in \{1,...,K_u\}$ hinsichtlich ihres Aufbaus aus atomaren Attributausprägungen vollständig expliziert. In den zugeordneten Ausdrücken $<m_{s(u.1).d},...,m_{s(u.Ku).d}>$ werden die Terme $m_{s(u.k).d}$ dagegen nicht expliziert. Dieser Zusammenhang wird anhand eines einstelligen Prädikatssymbol und einer Attributmarke erhellt:

$$mu_{u.n.d} \bullet <m_{s(u).d}> \approx mu_{u.n.d} \bullet prä_u(m_{s(u).d}{}^{ex})$$

$:\Leftrightarrow$ att_marke$_{s(u)}$, OB_{att_marke}=SYMBOL

$Prä_u(att\_marke_{s(u)})$

$Marke_{j(s(u))}$: $attribut_{q(j(s(u)).1)} ... attribut_{q(j(s(u)).Kj(s(u)))} \rightarrow att\_marke_a$

$marke_{j(s(u))}$: $OB_{att.q(j(s(u)).1)} x...x\, OB_{att.q(j(s(u)).Kj(s(u)))} \rightarrow OB_{att\_marke}$

$(at_{a.1},...,at_{a.Ka}) \rightarrow marke_{j(s(u))}(at_{s(u).1},...,at_{s(u).Ks(u)})=m_{s(u)}$

$m_{s(u).d}{}^{ex} = marke_{j(s(u))}(at_{s(u).1.d},...,at_{s(u).Ks(u).d})$

Daher gilt:

$$mu_{u.n.d} \bullet <m_{s(u).d}> \approx mu_{u.n.d} \bullet prä_u(m_{s(u).d}{}^{ex})$$
$$\wedge\ \mu_{s(u).d}{}^{ex} = marke_{j(s(u))}(at_{s(u).1.d},...,at_{s(u).Ks(u).d})$$

$$\Leftrightarrow\ mu_{u.n.d} \bullet <m_{s(u).d}> \approx mu_{u.n.d} \bullet prä_u(marke_{j(s(u))}(at_{s(u).1.d},...,at_{s(u).Ks(u).d}))$$

33) Auch hier dient ein einfaches Beispiel zur Verdeutlichung. Er erstreckt sich wiederum auf ein einstelliges Prädikatssymbol und eine Attributmarke:

$$mu_{u.0.d} \bullet <m_{s(u).d}> \approx fakt_0(mu_{u.0.d}, prä_u(m_{s(u).d}{}^{ex}))$$

$:\Leftrightarrow$ att_marke$_{s(u)}$, OB_{att_marke}=SYMBOL

$Prä_u(att\_marke_{s(u)})$

$fakt_0(mu_{u.0.d}, prä_u(m_{s(u).d}{}^{ex}))$

$Marke_{j(s(u))}$: $attribut_{q(j(s(u)).1)} ... attribut_{q(j(s(u)).Kj(s(u)))} \rightarrow att\_marke_a$

$marke_{j(s(u))}$: $OB_{att.q(j(s(u)).1)} x...x\, OB_{att.q(j(s(u)).Kj(s(u)))} \rightarrow OB_{att\_marke}$

$(at_{a.1},...,at_{a.Ka}) \rightarrow marke_{j(s(u))}(at_{s(u).1},...,at_{s(u).Ks(u)}) = m_{s(u)}$

$m_{s(u).d}{}^{ex} = marke_{j(s(u))}(at_{s(u).1.d},...,at_{s(u).Ks(u).d})$

Daher gilt:

$$mu_{u.0.d} \bullet \langle m_{s(u).d} \rangle \approx fakt_0(mu_{u.0.d}, prä_u(m_{s(u).d}{}^{ex}))$$
$$\wedge\ m_{s(u).d}{}^{ex} = marke_{j(s(u))}(at_{s(u).1.d}, \ldots, at_{s(u).Ks(u).d})$$

$$\Leftrightarrow\quad mu_{u.0.d} \bullet \langle m_{s(u).d} \rangle \approx fakt_0(mu_{u.0.d}, prä_u(marke_{j(s(u))}(at_{s(u).1.d}, \ldots, at_{s(u).Ks(u).d})))$$

34) Für $FAK_{u.0} \neq \emptyset$ existiert mindestens ein solches Fakt, d.h. es gilt $D_u \in \mathcal{N}_+$. Andernfalls ist für die leere Ausgangsfaktenmenge $FAK_{u.0} = \emptyset$ mit $D_u=0$ überhaupt kein multiples Fakt definiert, das in der Sektion "Fakten" für das Prädikatssymbol $Prä_u$ aufgelistet werden könnte. Falls die gesamte Ausgangsfaktenmenge mit $FAK_0 = \emptyset$ für alle Prädikatssymbole leer ist, wird dies in der Sektion "Fakten" durch die Notation "---" angezeigt.

35) Dies sind die "wesentlichen" Aspekte, auf die in der eingangs angeführten Festlegung der speziellen Netzdarstellung Bezug genommen wurde.

36) Der Begriff der perzeptiven Überfrachtung ist nach Einschätzung des Verf. nicht wohldefiniert. Es liegt auch außerhalb des Erkenntnisinteresses dieser Arbeit, diesen Begriff inhaltlich zu präzisieren. Er wird hier nur als unscharfe Umschreibung für die Erfahrung benutzt, daß Graphiken tendenziell um so mehr an Verständlichkeit verlieren, je größer die Anzahl der graphikkonstituierenden Elemente und je komplexer ihre graphisch ausgedrückten Beziehungen untereinander werden.

37) Gegenüber dem Begriff des mittleren Komplexionsniveaus konzidiert der Verf. Einschränkungen, wie sie bereits analog gegenüber dem Begriff perzeptiver Überfrachtung geäußert wurden.

38) Beispielsweise sieht der Verf. keinen erfolgversprechenden Ansatz, das allgemeine Übergangsschema oder die daraus konkretisierten Übergangsoperationen (Schaltregeln) als Aspekte der dynamischen Netzstruktur in einer graphischen Netzdarstellung verständlich auszudrücken. Ein konkretes Netzverhalten, das als Abfolge von Netzmarkierungen definiert ist, könnte zwar durch eine korrespondierende Folge von Netzgraphiken repräsentiert werden, deren jede genau eine Netzmarkierung darstellt. Doch würde bei größeren Netzen und ausgedehnteren Markierungsfolgen das Volumen der vielfachen Netzgraphiken derart aufgebläht, daß perzeptive Überfrachtung drohte.

39) Bei dieser Vereinfachung wird der Repräsentant mit dem Repräsentatum gleichgesetzt. Dies wird solange zugelassen, wie aus dem Argumentationskontext ersichtlich ist, daß nicht das algebraisch-prädikatenlogisch definierte Netz, sondern dessen graphische (Teil-)Repräsentation gemeint ist. Diese Vereinfachung wurde schon früher stillschweigend unterstellt, als die graphische Repräsentation von Stelle/Transition-Netzen erläutert wurde.
Strenggenommen stellen aber die repräsentierende Netzgraphik und die repräsentierten Aspekte eines Synthetischen Netzes unterschiedliche Konstrukte dar. Zwar läßt sich das Synthetische Netz als ein mathematischer Graph auffassen. Doch stellt dieser Graph ein abstraktes mathematisches Konstrukt dar. Er ist - trotz der irreführenden Äquivokation - grundsätzlich unabhängig von seiner konkreten "graphischen" Visualisierung durch eine Netzgraphik definiert. Denn mathematische Graphen lassen sich ohne jede "graphische" Darstellung als formalsprachliche Konstrukte ausdrücken und behandeln. Daher besteht grundsätzlich ein Unterschied zwischen mathematisch definierten Graphen und ihren graphischen Visualisierungen. Letzte werden daher auch als visualisierte Graphen angesprochen, um sie von den zugrundeliegenden mathematischen Graphen abzuheben.
Zwecks Vereinfachung der Terminologie wird jedoch in dieser Arbeit ebenso erlaubt, auf die korrekte Differenzierung zwischen mathematischen und visualisierten Graphen zu verzichten. Gleiches gilt für die Unterscheidung zwischen den Komponenten von mathematischen Graphen einerseits und den korrespondierenden Komponenten aus ihren graphischen Visualisierungen andererseits. Daher werden z.B. die Repräsentationen von Knoten und Kanten, die ursprünglich in einem zu repräsentierenden mathematischen Graphen definiert sind, auch in der graphischen Darstellung des Graphen als Knoten und Kanten angesprochen. Ohne die zuvor eingeräumte Vereinfachung müßte dagegen von kreisförmigen oder rechteckigen Knotenrepäsentanten bzw. von pfeil- oder linienförmigen Kantenrepräsentanten gesprochen werden.

40) Der Verf. läßt diese Redundanzoption zu, damit bei der Gestaltung von Netzmodellen der jeweils agierende Modellierungsträger den Freiheitsgrad behält, selbst über das erwünschte Ausmaß kontrollierter Redundanz zu entscheiden.

41) Allerdings gilt die Kombination aus Netzgraphik und -legende nur für genau einen Netzzustand. Daher muß die semi-graphische Netzrepräsentation modifiziert werden, sobald der Netzzustand verändert worden ist. Diese Anpassung bleibt zumeist nicht auf die veränderte Faktenmenge in der Netzlegende beschränkt. Vielmehr muß in der Regel auch die Netzgraphik neu erstellt werden, um die veränderte Netzmarkierung korrekt zu repräsentieren. Falls hierbei die Markenkopien in ihrer impliziten Kurznotation $m_{s.d}$ dargestellt werden, kann eine weitere Anpassung in der Netzlegende erforderlich werden. Dies ist immer dann der Fall, wenn sich nicht nur die Verteilung der Markenkopien über die Stellen des Netzes verändert hat, sondern auch Modifizierungen der Attributausprägungen strukturierter (und vollständig explizierter) Markenkopien eingetreten sind.
Solche Anpassungserfordernisse bereiten jedoch erheblichen Aufwand und können leicht dazu führen, daß der Benutzer eines Netzmodells den Überblick verliert. Daher werden solche Anpassungen in dieser Arbeit grundsätz-

lich nicht weiter verfolgt. Statt dessen wird vorausgesetzt, daß die semi-graphischen Netzdarstellungen jeweils nur für genau einen Netzzustand "r" benutzt werden. Die Abfolge aufeinander folgender Netzzustände wird ausschließlich mit dem präzise und übersichtlich definierten Konzept Synthetischer Netze für das Transformieren von Netzmarkierungen und Faktenmengen durch das Schalten von Transitionen studiert. Erst nachdem innerhalb des formalen Apparats Synthetischer Netze eine neue Netzmarkierung bzw. Faktenmenge abgeleitet worden ist, wird dieser neue Netzzustand in seine semi-graphische Darstellungsweise umgesetzt. Diese Netzdarstellung wird nicht aus Anpassungen der Repräsentationen früherer Netzzustände gewonnen, sondern unmittelbar aus dem jeweils neu eingetretenen Netzzustand abgeleitet.

42) Die Möglichkeit, in der Sektion "Fakten" Faktenmengen für verschiedene Netzzustände zu repräsentieren, entspricht genau der früher erhobenen und gerechtfertigten Forderung, für prädikatenlogische Objektmodelle zeitlich variable Faktenmengen erfassen zu können. Dieser Ansatz wird später bei der PROLOG-Implementierung von Netzmodellen derart fortgeführt, daß die Sektion "Fakten" als eine dynamische Datenbasis realisiert wird.

43) Daher umfaßt die PROLOG-Implementierung von Netzmodellen, die in der voranstehenden Anmerkung erwähnt wurde, den zustandsidentifizierenden Index "r" nicht mehr: Dort werden die faktischen Formeln $fakt_r(mu_{u.r}, prä_u(m_{s(u.1)},...,m_{s(u.Ku)})$ für einen beliebigen, aber jeweils für alle Formeln gleichen Netzzustand "r" nur als Fakten $fakt(mu_u,prä_u(m_{s(u.1)},...,m_{s(u.Ku)})$ in einer dynamischen Datenbasis dargestellt. Der aktuelle Zustand der Datenbasis entspricht jeweils genau einem Netzzustand "r", ohne diesen explizit anzugeben. Auf diesen Explizierungsverzicht von implementierungsbezogenen Netzkonstrukten wird später nicht mehr ausdrücklich hingewiesen. Er wird fortan als bekannt vorausgesetzt.
Über die Vernachlässigung des zustandsidentifizierenden Index "r" bei der PROLOG-Implementierung von Netzmodellen kann durchaus gestritten werden. Er bedeutet einerseits einen Informationsverlust. Andererseits erlaubt er, die aufwendige Verwaltung von indexrepräsentierenden Informationen einzusparen, die ohnehin für alle faktischen Formeln desselben Zustands einer dynamischen Datenbasis gleich ausfallen (sollen). Dabei erstreckt sich der Verwaltungsaufwand nicht nur auf die reine Speicherung der indexrepräsentierenden Informationen. Vielmehr muß auch dafür gesorgt werden, daß die Zustandsindizes "r" aller faktischen Formeln in der Datenbasis korrekt aktualisiert werden. Andernfalls kann es leicht zu inkonsistenten Formelindizierungen kommen. Denn durch das Schalten einer Transition werden nur diejenigen faktischen Formeln unmittelbar betroffen, die in der transitionszugehörigen Transaktion als zu entfernende oder als zu ergänzende Formelvorkommnisse explizit aufgeführt werden. Falls nur die Indizes dieser unmittelbar betroffenen Formelvorkommnisse aktualisiert würden, befände sich die Datenbasis nach dem Schalten einer Transition in einem inkonsistenten Zustand: Die unmittelbar betroffenen Formelvorkommnisse besäßen den korrekten Index "f" der erzeugten Folgefaktenmenge FAK_f. Aber alle anderen faktischen Formeln, die vom Schalten der Transition nicht betroffen wurden, wiesen noch den Index "r" der unmittelbar vorangehenden Referenzfaktenmenge FAK_r aus. Dann bezöge sich *derselbe* Zustand der dynamischen Datenbasis zwar inhaltlich auf den neuen Netzzustand mit der Folgefaktenmenge FAK_f. Doch enthielte sie faktische Formeln, die aufgrund ihrer Indizierungen zu *verschiedenen* Netzzuständen zu gehören scheinen. Dies wäre inkonsistent. Eine Indexverwaltung müßte daher sicherstellen, daß solche Integritätsverletzungen der dynamischen Datenbasis nicht auftreten können. Entsprechende integritätswahrende Mechanismen sind seitens der Automatischen Informationsverarbeitung durchaus bekannt, bereiten aber erheblichen Aufwand. Vermutlich unterblieb (u.a.) aus diesem Grund im PROLOG-basierten Programmpaket PASIPP, das in dieser Arbeit der Implementierung von Netzmodellen zugrundeliegt, die zustandsspezifische Indizierung der faktischen Formeln von vornherein.
Statt dessen werden alle faktischen Formeln, die zum selben Netzzustand "r" gehören, in einer zustandsspezifischen PROLOG-Datei zusammengefaßt. Der Dateiname verweist dabei auf den zugehörigen Netzzustand. Z.B. können alle faktischen Formeln $fakt_r(mu_{u.r},prä_u(m_{s(u.1)},...,m_{s(u.Ku)})$ zunächst als Formeln $fakt(mu_u,prä_u(m_{s(u.1)},...,m_{s(u.Ku)})$ implementiert und dann in einer zustandsspezifischen Datei mit dem Namen "FAKTEN_R.DAT" gespeichert werden. Auf diese Weise wird die Information über den Netzzustand "r", zu dem die faktischen Formeln gehören, im Dateinamen indirekt repräsentiert, ohne ihn durch eine entsprechende Formelindizierung explizit auszuweisen. In die dynamische Datenbasis wird vom Programmpaket PASIPP jeweils diejenige Datei "FAKTEN_R.DAT" geladen, die alle faktischen Formeln für den aktuellen Netzzustand "r" umfaßt. Auf diese Weise wird die Integrität der dynamischen Datenbasis bei der schaltbedingten Aufeinanderfolge verschiedener Netzzustände gewahrt, ohne eine explizite - aber infolgedessen aufwendige - Formelindizierung verwalten zu müssen. Vgl. zu dieser Verwaltung von zustandsspezifischen Dateien für Fakten- oder Markierungsmengen durch das Programmpaket PASIPP OBERWEIS (1988a), S. 6f., 11f; OBERWEIS (1988b), S. 303; OBERWEIS (1988c), S. 6f., 12f. u. 19. In einer neueren Fortenwicklung von PASIPP wurde allerdings der ober skizzierte Ansatz wieder aufgegeben. Dort scheint eine zustandsspezifische Indizierung der faktischen Formeln zu geschehen; vgl. OBERWEIS (1989a), S. 14. Allerdings ist die Implementierungsvariante in den vorgenannten Quellen so knapp skizziert, daß sich ihr die exakte Behandlung von zustandsspezifischen Formelindizes nicht entnehmen läßt.

44) Das schließt einerseits alle zulässigen Netzzustände ein, die sich von der Ausgangsmarkierung des Netzes aus durch Ausführungen von Schaltschritten erreichen lassen. Andererseits gehören dazu ebenso alle unzulässigen Netzzustände. Sie können zwar nicht in der vorgenannten Weise erreicht, wohl aber durch Faktenmengen spezifiziert werden. Auf diese Weise ist es möglich, in der Netzlegende - zusammen mit der Repräsentation von Fakten durch

Markenkopien in der Netzlegende - auch unzulässige Sachverhalte zu thematisieren. In dieser letzten Hinsicht ist die hier gewählte Darstellungsweise von Netzen der Verwendung von Erreichbarkeitsgraphen überlegen. Denn dort werden nur zulässige Netzzustände als erreichbare Markierungen erfaßt.

45) In dieser Hinsicht besteht eine enge Verknüpfung zum "frame"-Problem, das in einer früheren Anmerkung skizziert wurde. Denn durch den Übergang zwischen einer Referenzfaktenmenge FAK_r und ihrer Folgefaktenmenge FAK_f ändern sich oftmals nur wenige der faktischen Formeln. Dennoch müssen in den zugehörigen Subsektionen jeweils die *gesamten* Faktenmengen aufgeführt werden. Dies bedeutet einerseits, daß die Netzlegende keine "intelligente" Lösung für das "frame"-Problem bietet. Andererseits resultiert daraus auch die nachfolgend monierte Aufwendigkeit und Unübersichtlichkeit einer Netzlegende, in der die Faktenmenge von *mehreren* Netzzuständen aufgeführt werden.

46) Für die vollständige Spezifizierung der Faktenmengen aller zulässigen Netzzustände eignet sich eher die Verwendung von Erreichbarkeitsgraphen. In ihnen werden alle zulässigen Netzzustände als erreichbare Netzmarkierungen ausgewiesen. Darüber hinaus explizieren sie den dynamischen Zusammenhang der zulässigen Netzzustände, indem sie deren wechselseitige Erreichbarkeit durch Ausführungen von Schaltschritten repräsentieren.

47) Dabei handelt es sich im wesentlichen um die Kleinschreibung von Operations- und Prädikatssymbolen in PROLOG-Programmen sowie um die Ersetzung des ":" zwischen den Attributsorten und ihren Objektmengen durch ein "=". Darüber hinaus werden Teile der Stellen- und Faktendeklarationen aus ihren Hauptbereichen "predicates" bzw. "database" in den Bereich "clauses" ausgelagert.

48) Dies betrifft im wesentlichen die PROLOG-Bereiche "domains" für Marken und Operationssymbole, "predicates" für Stellen und Prädikatssymbole, "clauses" für Transitionen und Transaktionen sowie "database" für Fakten. Hinzu kommt ein informationstechnisch bedingter Bereich "goal" für die Initialisierung der dynamischen Datenbank "database" mit den Fakten der Ausgangsmarkierung.

49) Daher wurde hier auch die Faktenanzahl gegenüber dem Original reduziert, um die Beispieldarstellung auf ihren wesentlichen Kern zu beschränken. Nebenbei wird darauf hingewiesen, daß die Einfachheit des Beispiels zu der seltenen Ausnahme führt, daß die lokal definierte Übergangsoperation der Transaktion tr_1, die der Transition t_1 zugeordnet ist, hier zugleich globalen Charakter trägt. Denn bei nur einer existierenden Transaktion fallen lokale und globale Operationsdefinitionen zusammen.

50) Vgl. OBERWEIS (1988a), S. 5ff.; OBERWEIS (1988c), S. 6ff. u. 13. Geringfügige Modifizierungen wurden vorgenommen, um das Beispiel realitätsnäher zu gestalten. Die Notierung der Kraftfahrzeug-Kennzeichen wurde verändert und der Test des Mindestalters von "Alter>21" zu "Alter≥21" korrigiert. Die Schreibweise des Originals, die an der PROLOG-Syntax ausgerichtet ist, wurde hier an die Sortenschreibweise des Signaturkonzepts angepaßt. Dies bedeutet die Eliminierung der "," bei Sortenaufzählungen in Argumenten von Operations- und Prädikatssymbolen. Hinzu kommt die differenzierende Groß- oder Kleinschreibung je nachdem, ob Operations- oder Prädikatssymbole bzw. Operationen oder prädikatenlogische Formen vorliegen. Außerdem wurde die Notation von Umlauten an die deutsche Orthographie angepaßt.
Ein Vergleich der Beispielrepräsentation, wie sie in der vorgenannten Quelle durchgeführt wird, mit der hier präsentierten Darstellungsweise ließe sich benutzen, um Besonderheiten Synthetischer Netze herauszustellen. Ein systematischer Vergleich zwischen konventionell definierten Prädikat/Transition-Netzen und Synthetischen Netzen liegt jedoch nicht im Erkenntnisinteresse dieser Arbeit. Der Verf. wird nachfolgend nur auf einige exemplarisch ausgewählte, besonders auffällige Abweichungen der beiden Repräsentationsweisen aufmerksam machen.

51) Im Original wird aufgrund der dort verwandten PROLOG-Dialekte ein anderes Begrenzungszeichen (') benutzt. Hier wird sofort das Begrenzungszeichen (") der Programmiersprache Turbo-PROLOG mit gewählt, weil sie der präsentierten Ausarbeitung zugrundeliegt.

52) Es kommt zwar der Ausdruck "marke" vor. Doch bezieht er sich nicht auf Marken im hier eingeführten Sinn. Statt dessen dient er nur dazu, die Netzmarkierung durch Fakten zu notieren. Daher entspricht er dem Ausdruck "$fakt_0$" aus der Darstellungsweise des Verf.

53) Vgl. dazu die Deklaration des Prädikats "kunde_mit_vertrag" bei OBERWEIS (1988a), S. 5f.: Dort wird bezüglich ges gemieteten Wagens nur die Information über dessen Attribut "kfz_nr" bewahrt. Sollte später - etwa bei einer Beschädigung des Mietwagens durch den Kunden - das Baujahr des Wagens von Interesse sein, so träte die unerwünschte Fernwirkung ein, daß aus der Verbindung des Kunden mit dem gemieteten Wagen allein das Baujahr nicht mehr ersichtlich wäre.

54) Es wird darauf verzichtet, die Menge IB aller netzkonstitutiven Integritätsbedingungen explizit zu notieren. Gleiches gilt für und das allgemeine Übergangsschema ÜS, die Variablenfamilie VAF, die teilevaluierte Termfamilie TTMF(VAF) und die Menge VBM aller zulässigen Variablenbelegungen vb_c. Diese fakultativen Konstituenten der Netzdefinition gelten für alle Synthetischen Netze in derselben Weise. Sie werden als implizit vereinbart vorausgesetzt. Dabei wird auf die frühere schematische Definition von Synthetischen Netzen Bezug genommen.

5.1.3 Anschluß Synthetischer Netze an die Objektmodellierung

5.1.3.1 Schnittstellen zur Modellimplementierung

Für die Modellierung von Koordinierungsproblemen bei Flexiblen Fertigungssystemen werden in dieser Arbeit Netzmodelle verwendet[1]. Sie erfüllen das allgemeine Definitionsschema Synthetischer Netze und dessen spezielle semi-graphische Darstellungsweise, die in den voranstehenden Kapiteln entwickelt wurden. Synthetische Netze fallen trotz ihrer bereits vereinfachten Notationsweise noch relativ[2] komplex aus. Daher erfordert ihre praktische Anwendung im Regelfall, die Netzmodelle mit der Hilfe von Automatischen Informationsverarbeitungssystemen zu implementieren.

Die Netzimplementierung wird nachfolgend nur so weit beschrieben, daß sie für jedes konkret definierte und semi-graphisch dargestellte Netz ausgeführt werden kann. Dabei wird die grundsätzliche Implementierungsmöglichkeit durch eine allgemeingültige Implementierungsmethode aufgewiesen. Es liegt nicht mehr im Erkenntnisinteresse dieser Arbeit, die Netzimplementierungen in Einzelfällen zu verwirklichen. Es würde sich um eine systematische Methodenanwendung handeln. Sie bereitete zwar erheblichen Darstellungsaufwand, ließe aber keinen bemerkenswerten Erkenntnisgewinn für das netzgestützte Modellierungskonzept erwarten.

Implementierungsgrundlage ist das Softwarepaket PASIPP[3]. Es wurde bereits an früherer Stelle als ein informationstechnisches Fundament dieser Ausarbeitungen herausgestellt. Seine Konstrukte werden hier als bekannt vorausgesetzt[4]. Sie werden nur in dem Ausmaß thematisiert, wie sie für die Implementierung spezieller Aspekte von Synthetischen Netzen besondere Bedeutung erlangen oder zu diesem Zweck speziellen Modifizierungen unterworfen werden. Die Implementierungssoftware beruht auf der Programmiersprache PROLOG. Ausdrucksmächtigkeit und Inferenzpotential dieser Programmiersprache wurden schon ausführlicher behandelt. Daher wird auch auf diese Aspekte hier nicht weiter eingegangen.

Allerdings unterliegt die PROLOG-Basis des Programmpakets PASIPP einer grundsätzlichen Veränderung. Es stand dem Verf. in drei speziellen PROLOG-Dialekten[5] zur Verfügung, die dem algebraischen Signaturkonzept nicht vollauf gerecht werden. Mit ihrer Hilfe ist es nicht möglich, prädikatenlogische Formeln mit überlagerten Sortenstrukturen unmittelbar auszudrücken. Daher hat der Verf. seinen Ausführungen einen anderen PROLOG-Dialekt - die Programmiersprache Turbo-PROLOG - zugrundegelegt[6]. Sie gestattet die unmittelbare Formulierung von sortierten Prädikatssymbolen. Darüber hinaus zeichnet sie sich durch eine Segmentierung der PROLOG-Programme[7] aus, die in exzellenter Weise den oben vorgestellten Sektionen aus der semi-graphischen Darstellung von Synthetischen Netzen entspricht[8].

Die nachfolgend präsentierte Methode zur Implementierung von Synthetischen Netzen knüpft unmittelbar an die semi-graphische Darstellungsweise von Netzen an. Sie beachtet bereits alle Syntaxkonventionen der Programmiersprache Turbo-PROLOG[9]. Die Methodenanwendung liefert daher Turbo-PROLOG-Programme, die ohne weitere Überarbeitungen unmittelbar auf einem Automatischen Informationsverarbeitungssystem implementiert werden können. Aufgrund ihrer gesicherten Implementierungsfähigkeit werden diese Programme kurz als Netzimplementierungen bezeichnet[10]. Sofern die derart implementierten[11] Synthetischen Netze reale Objekte modelliert haben, wird auch von implementierten Netzmodellen gesprochen.

Eine entsprechende Anpassung des Softwarepakets PASIPP an den Dialekt Turbo-PROLOG erfolgt hier nicht[12]. Eine entsprechende Reimplementierung würde nur erheblichen Aufwand bereiten, ohne grundsätzlich neuartige Erkenntnisse für das Modellierungskonzept zu verheißen. Daher wird diese rein technische Aufgabe der Dialekttransformation nicht weiter berücksichtigt[13].

Transformationsmethode

Ausgangspunkt der Transformationsmethode ist die semi-graphische Repräsentation eines beliebigen Synthetischen Netzes durch eine Netzgraphik und eine Netzlegende. Ziel ist es, diese Netzdarstellung in ein Turbo-PROLOG-Programm zu transformieren, das sich in einem Automatischen Informationsverarbeitungssystem unmittelbar implementieren und anschließend ausführen läßt. Die Transformationsmethode stellt eine endliche Sammlung von Implementierungsvorschriften dar. Hierfür gilt[14)]:

a) Die Programmiersprache Turbo-PROLOG unterscheidet - wie auch alle anderen etablierten PROLOG-Dialekte - nicht zwischen Prädikats-, Funktions- und Konstantensymbolen einerseits sowie prädikatenlogischen Formeln, Funktionen bzw. Konstanten andererseits. Es wird durchgängig nur von Prädikaten, Funktionen und Konstanten gesprochen. Diese werden stets mit einleitendem Kleinbuchstaben notiert im Gegensatz zu Variablen, die immer mit einem Großbuchstaben beginnen müssen. Es ist nur aus dem jeweils betroffenen Programmkontext ersichtlich, ob die klein notierten "Prädikate", "Funktionen" oder "Konstanten" die Qualität von Prädikats-, Funktions- bzw. Konstantensymbolen besitzen[15)]. Um die notationellen und terminologischen Turbo-PROLOG-Konventionen einzuhalten, wird fortan nur noch von Prädikaten, Funktionen und Konstanten gesprochen[16)].

b) Indizierungen lassen sich in Turbo-PROLOG nicht als Subskripte darstellen. Daher werden die Indizes aus der früheren Netzdefinition und aus der semi-graphischen Netzrepräsentation in PROLOG-Programmen auf drei Weisen behandelt:

- ❑ Die Indizes können fortgelassen werden, falls sie zur Erzeugung eines wohldefinierten PROLOG-Programms nicht erforderlich sind.
- ❑ Andernfalls lassen sich die Indizes an die jeweils zu indizierenden Ausdrücke subskriptfrei anhängen. Hierfür kommen zwei Alternativen der Postfix-Bildung in Betracht:
  - ➥ Der Index wird an den indizierten Ausdruck unmittelbar angehängt. Aus der Sortennotation "attribut_7" wird die PROLOG-Schreibweise "attribut7".
  - ➥ Zwischen den angehängten Index und den indizierten Ausdruck tritt das Trennungszeichen "_". Die Sortennotation "attribut_q" wird in die PROLOG-Schreibweise "attribut_q" transformiert.

Die letzte (vorletzte) Alternative wird bei Indizes verwendet, die aus Buchstaben (Ziffern) aufgebaut sind. Die erste Alternative betrifft vor allem Faktenindizes "r", die auf den jeweils aktuellen Modellzustand verweisen. Solche Indizierungen unterbleiben, weil die Sammlung aller Fakten in einer dynamischen PROLOG-Datenbank repräsentiert stets den aktuellen Zustand "r" eines implementierten Netzmodells, ohne daß dieser Index "r" explizit notiert würde.

c) Sorten werden in der Sprache Turbo-PROLOG nicht immer als solche expliziert. Oftmals werden sie nur als Namen[17)] oder Objekte[18)] angesprochen: Dann erfolgt keine klare Abgrenzung gegenüber anderen Namen - etwa Prädikats- oder Funktionsnamen - und in bezug auf andere Objekte, wie z.B. Konstanten. Mitunter werden die Sorten jedoch auch korrekt als objektsprachlicher Ausdruck "domain" von anderen Syntaxkomponenten abgegrenzt[19)]. Diesem Ansatz verfolgt der Verf. aufgrund seiner konzeptionellen Klarheit. Darüber hinaus korrespondiert diese Sortenbezeichnung genau mit dem Namen desjenigen Programmbereichs, in dem die Sorten definiert werden: der Sektion "domains".

d) Von der Programmiersprache Turbo-PROLOG werden fünf Standardsorten mit zugehörigen Mengen zulässiger formaler Objekte vorgehalten[20]:

- Zur Sorte "symbol" mit der Objektmenge OB_{symbol}=SYMBOL gehören alle symbolischen Ausdrücke, die in der formalsprachlichen PROLOG-Syntax für die Argumente von Prädikats- und Funktionssymbolen überhaupt gebildet werden können[21]. Die formalen Objekte aus der Menge SYMBOL können atomar oder auch zusammengesetzt sein. Sie stellen aus der Perspektive des Signaturkonzepts Terme dar. Es handelt sich um eine global definierte Sorte. Sie umfaßt alle nachfolgend angesprochenen Standardsorten und auch alle speziell definierbaren Nonstandardsorten als Subsorten. Daher wird die globale Sorte "symbol" immer dann gewählt, wenn keine andere, näher spezifizierte Sorte bekannt ist.
- Die Sorte "char" umfaßt in ihrer Objektmenge OB_{char}=CHAR alle Normalzeichen[22] des ASCII-Codes[23]. Jedes einzelne Zeichen <z> wird in Turbo-PROLOG durch die Schreibweise 'z' gekennzeichnet.
- Die Sorte "string" enthält in ihrer Objektmenge OB_{string}=STRING alle endlichen Zeichenfolgen (strings), die aus Zeichen der Sorte "string" gebildet werden können. Solche Zeichenfolgen wurden schon früher als Signatur "STRING" und zugehörige Algebra eingeführt. Jede Zeichenfolge <string> wird in Turbo-PROLOG als "string" notiert.
- Die Sorte "integer" erstreckt sich auf die Objektmenge $OB_{integer}$=INTEGER, die alle Ganzzahlen umfaßt[24].
- Die Sorte "real" umgreift in ihrer Objektmenge OB_{real}=REAL alle reellen Zahlen[25].

e) Jede Attributsorte, die in originärer Weise als eine der fünf PROLOG-Standardsorten definiert ist, wird durch das metasprachliche Zeichen "=" unmittelbar mit dieser Sorte identifiziert. Eine Objektmengenangabe entfällt. Die sortenspezifische Objektmenge gilt im Sinne der oben erfolgten Zuordnungen als implizit vereinbart. Beispielsweise wird eine Attributsorte "$attribut_q$", deren zulässigen Ausprägungen Ganzzahlen aus der Objektmenge OB_q=INTEGER darstellen, in der PROLOG-Implementierung durch "attribut_q=integer" notiert.

f) Mit der Hilfe von Konstanten werden die Mengen zulässiger formaler Objekte für originär definierte Nonstandardsorten festgelegt, die nicht mit einer von den Standardsorten zusammenfallen[26]. Da jede Konstante ebenso als nullstellige Funktion aufgefaßt werden kann[27]), lassen sich die Objektmengen auch aus nullstelligen Funktoren aufbauen. Beispielsweise kann die Objektmenge $OB_{bas\_marke}=\{\emptyset\}$ der Basismarkensorte "bas_marke" mit dem Erzeugungs-Operationssymbol "$Marke_0$" in Turbo-PROLOG sowohl mit der Konstante "marke_0" als auch mit dem nullstelligen Funktor "marke_0()" ausgedrückt werden: OB_{bas_marke}={marke_0}= {marke_0()}. Allerdings werden - wie schon bei den originär definierten Standardsorten - die Objektmengen nicht direkt angegeben. Vielmehr werden die formalen Objekte, die zur Objektmenge einer originär definierten Nonstandardsorte gehören, hinter dem Namen der Sorte und einem Gleichungszeichen aufgelistet. Falls es sich um mehrere Objekte handelt, werden sie durch ein Semikolon (";") voneinander getrennt[28]. Beispielsweise wird die Sorte der Basismarke notiert als: bas_marke=marke_0 oder bas_marke=marke_0()[29]. Entsprechend wird eine Attributmarke der Sorte "attribut_q" mit den zwei zulässigen Ausprägungen "positiv" und "negativ" in der Netzimplementierung dargestellt als[30]: attribut_q=positiv; negativ oder attribut_q= positiv(); negativ().

g) Weitere Nonstandardsorten können als Zielsorten von Funktionen[31] aus den originär definierten Standard- und Nonstandardsorten in beliebig komplexer Weise induktiv zusammengesetzt werden[32]. Dadurch werden die weiterführenden Nonstandardsorten derivativ definiert[33]. Z.B. wird das Attribut "$attribut_2$" in einer Netzlegende durch "$attribut_2$=Att($attribut_1$)" als Zielsorte des Operationssymbols "Att" derivativ bestimmt; ihr liegt die originär definierte Sorte "$attribut_1$" mit der Objektmenge $OB_{att.1}$=SYMBOL zugrunde. Dann wird dies in der Netzimplementierung durch die PROLOG-Konstrukte "attribut2=att(attribut1)" und "attribut1=symbol" ausgedrückt.

h) Dieselbe Nonstandardsorte kann multipel definiert sein[34]. Wenn dies der Fall ist, werden zunächst der Sortenname und ein Gleichungszeichen ("=") genau einmal notiert. Die Definientes aus den alternativen Definitionen derselben Sorte werden danach aufgelistet und jeweils durch ein Semikolon (";") voneinander getrennt. Beispielsweise wird die Markensorte "att_marke" rein derivativ, aber in multipler Weise als Zielsorte der beiden Operationssymbole "Marke_1" und "Marke_2" mit Marke_1: $\text{attribut}_{q(1)} \rightarrow \text{att\_marke}$ bzw. Marke_2: $\text{attribut}_{q(2)} \rightarrow \text{att\_marke}$ und $q(1) \neq q(2)$ definiert. Dies wird in Turbo-PROLOG notiert als: att_marke=marke1(attribut_q(1)); marke2(attribut_q(2)).

i) Alle Nonstandardsorten werden im Programmbereich "domains" definiert[35]. Die Standardsorten der Programmiersprache Turbo-PROLOG werden nicht expliziert, weil sie in jeder Implementierung dieser Sprache implizit enthalten sind[36]. Die Gesamtheit der expliziten und impliziten Sortendeklaration entspricht der Sektion "Marken/Operationssymbole" aus der Netzlegende für die semi-graphische Darstellung Synthetischer Netze. Wenn von den implizit vorgehaltenen Standardsorten abstrahiert wird, liegt sogar eine unmittelbare Entsprechung der Sektionen "domain" und "Marken/Operationssymbole" vor.

j) Im Programmbereich "predicates" werden alle Prädikatssymbole Prä_u aus der Sektion "Stellen/Prädikatssymbole" als Prädikate prä_m mit m=u[37] aufgeführt[38]. Die Stellen s_m, die diesen Prädikatssymbolen mit $\text{bsp}(s_m)=\text{Prä}_u$ und $m \in \{1,...,M\}$ zugrundeliegen, werden nicht mehr angegeben. Sie gehen mittelbar in die spezielle PROLOG-Darstellung von Transaktionen ein.

k) Der Programmbereich "predicates" enthält auch das Kapazitätsprädikat "kapazitaet(prädikatsname,anzahl)". Es dient der Deklaration der stellenspezifischen Markenkapazitäten, die in der semi-graphischen Netzdarstellung innerhalb der Sektion "Stellen/ Prädikatssymbole" durch Formeln der Art "$\text{markenkapazitaet}_m=\text{KAP}_m$" spezifiziert werden. Die Sorten des Prädikatsarguments sind in der Sektion "domains" durch anzahl=integer und prädikatsname=symbol definiert.

l) Die konkreten Markenkapazitäten KAP_m aller Stellen s_m mit $m \in \{1,...M\}$ werden im Programmbereich "clauses" als atomare Klauseln kapazitaet(prä_m,kap_m) notiert[39]. Dabei werden die Prädikatsnamen prä_m nicht als Prädikate, sondern als beliebige symbolische Ausdrücke behandelt. Für die Netzimplementierung gilt folgende Zuordnung:

- Der Prädikatssymbolname "Prä_m", der in der Netzlegende der Stelle s_m durch "s_m: Prä_m ..." zugeordnet ist, belegt in der Netzimplementierung die erste Stelle des Prädikats "kapazitaet": prä_m := Prä_m
- Die Markenkapazität KAP_m, die in der Netzlegende der Stelle s_m durch "$\text{markenkapazitaet}_m=\text{KAP}_m$" zugeordnet ist, belegt in der Netzimplementierung die erste Stelle des Prädikats "kapazitaet": kap_m := KAP_m[40].

m) Faktische Formeln $\text{fakt}_0(\text{mu}_{0.u.d},\text{prä}_u(m_{s(u.1).d},...,m_{s(u.Ku).d}))$ aus der Sektion "Fakten" in der Netzlegende werden im Programmbereich "database" durch die Deklaration des faktischen Prädikats "fakt" beschrieben[41]: fakt(anzahl,prädikatsvorkommnis). Dabei handelt es sich um einen Spezialfall der bereits erläuterten Definition von Prädikatssymbolen in der Sektion "predicates". Die Sorten, die das Argument des faktischen Prädikats "fakt" konstituieren, werden zusätzlich in der Sektion "domains" durch anzahl=integer und prädikatsvorkommnis=symbol definiert. Als formale Objekte der symbolischen Sorte "prädikatsvorkommnis" können später[42] alle Vorkommnisse $\text{prä}_u(m_{s(u.1).d},...,m_{s(u.Ku).d})$ von atomaren prädikatenlogischen Formeln eingesetzt werden, deren zugrundeliegenden Prädikatssymbole Prä_u in der Sektion "predicates" definiert sind[43].

Dabei werden die Prädikate $prä_u(m_{s(u.1).d},...,m_{s(u.Ku).d})$ durch die Programmiersprache PROLOG wie die Bilder einer Funktion mit dem Funktor $prä_u(...)$ behandelt[44]. Die Anzahl $mu_{0.u.d}$ identischer Vorkommnisse derselben atomaren prädikatenlogischen Formel $prä_u(m_{s(u.1).d},...,m_{s(u.Ku).d})$ stellt jeweils ein formales Objekt für die Sorte "anzahl" dar.

n) Die faktischen Formelvorkommnisse $fakt_0(mu_{0.u.d},prä_u(m_{s(u.1).d},...,m_{s(u.Ku).d}))$, die im Ausgangszustand "r=0" des zu implementierenden Synthetischen Netzes gültig sind, werden in der Sektion "clauses" als atomare Formeln fakt(mu_u.d,prä_u(m_s(u.1).d,...,m_s(u.Ku).d)) für alle unterschiedlichen faktischen Formelvorkommnisse mit differenzierenden Indizes $d \in DF_u$ und für alle Prädikatssymbole mit den Indizes $u \in \{1,...,U=M\}$ notiert[45].

o) Alle faktischen Formelvorkommnisse des originären Netzzustands "r=0" werden im Programmbereich "goal" durch das Prädikat "save" in einer Datenbasis mit dem Namen "fakten.pro"[46] gespeichert. Es handelt sich um eine dynamische Datenbasis[47]. Denn die PASIPP-spezifische Prädikate "entferne" und "einfuege", auf die anschließend näher eingegangen wird, bewirken eine Veränderung des Fakteninhalts dieser Datenbasis aufgrund des Schaltens von Transitionen[48]. Durch das "save"-Prädikat wird die dynamische Datenbasis mit der Ausgangsmarkierung $M_0 = FAK_0$ eines Synthetischen Netzes initialisiert.

p) Jede Transaktion aus der Sektion "Transitionen/Transaktionen" und ihre zugrundeliegende Transition werden in der Netzimplementierung im Programmbereich "clauses" als eine zusammengesetzte Klausel erfaßt. Allerdings fällt im PROLOG-Programm die Transaktionsdarstellung komplexer als in der zugrundeliegenden Legende einer semi-graphischen Netzdarstellung aus. Denn die transaktionsrepräsentierenden Klauseln des Softwarepakets PASIPP stellen Subjugatformeln dar, in denen Informationen über drei Aspekte des jeweils implementiertes Synthetischen Netzes zusammengefaßt werden. Es werden innerhalb derselben Formel implementiert:

- ❑ die Transaktionen aus der Netzspezifikation;
- ❑ das allgemeine Übergangsschema in seinen transaktionsspezifischen Konkretisierungen;
- ❑ die gesamte Netztopologie mit ihren Stellen, Transitionen und Kanten.

Jede dieser Formeln wird fortan als Transaktions-Klausel bezeichnet. Sie stellt jeweils eine transaktionsspezifische Schaltregel für das implementierte Synthetische Netz dar. Daher ist sie die Implementierung der schon früher erwähnten individualisierten Schaltregeln SR_v von Synthetischen Netzen.

q) Für den Implementierungskomplex einer Transaktions-Klausel gelten im Hinblick auf eine beliebige Transaktion tr_n, die im zugrundeliegenden Synthetischen Netz einer Transition t_n mit $btt(t_n) = tr_v$ und v=n zugeordnet ist, folgende Regeln für $v \in \{1,...,V=N\}$[49]:

- ❑ Die Prätestbedingungen $prätest(FAK_{u.r})$ für alle $u \in IIB_v$ entfallen. Da sie als Tautologien formuliert werden, bleiben sie für die Netzimplementierung irrelevant. Sie können das Schaltverhalten des implementierten Synthetischen Netzes nicht beeinflussen.
- ❑ Die Inklusionstests $FAK_{u.r} \geq vb_c(MTAE_{u.v})$ für alle $u \in IEB_v$ werden ebenfalls ausgelassen. Sie können sich zwar auf das Schaltverhalten eines implementierten Netzes dadurch auswirken, daß sie die Aktivierung der transaktionszugehörigen Transition entweder zulassen oder aber verbieten. Dennoch brauchen sie nicht explizit angeführt zu werden. Denn die PASIPP-spezifischen Transaktions-Klauseln in Verbindung mit dem kombinierten Unifizierungs- und Resolutionskonzept der Programmiersprache PROLOG sorgen dafür, daß die Übergangsoperation der Transaktion tr_v nur dann ausgeführt wird, wenn die Inklusionstests für alle Prädikatssymbole ihres Einflußbereichs erfüllt sind.

- Die Haupttestbedingungen haupttest($vb_c(MTAE_{u.v})$) werden für alle $u \in IEB_n$ durch Formeln aus der transaktionsspezifischen Restriktionsformelmenge $RES_v=\{for_{z(v.1)},...,for_{z(v.Hv)}\}$ implementiert. Die Argumente dieser Formeln beziehen sich jeweils auf die teilevaluierten atomaren Formelvorkommnisse aus den Kantengewichten $MTAE_{u.v}$ oder deren Komponenten. Die Variablenbindungsfunktionen vb_c entfallen dabei, weil das Unifizierungskonzept der Programmiersprache Turbo-PROLOG die sortentreue Variablenbindung automatisch realisiert.

- Die Bestimmungsgleichungen $vb_c(X_k)=vb_c(bestimme_k(X_h::X_h \in VA_u \; u \in IEB_v))$ werden für alle $X_k \in VA_u$ und für alle $u \in INB_v$ als Gleichungen X_k=bestimme_k(X_h:: X_h$\in VA_u \wedge u \in IEB_v$) implementiert. Auch diese Gleichungen werden als Formeln aus der Restriktionsformelmenge RES_v aufgefaßt. Die Bestimmungsfunktionen "bestimme_k" sind beliebige algebraische Funktionen. Die Variablenbindungsfunktionen vb_c entfallen wiederum aufgrund der PROLOG-Realisierung des Unifizierungskonzepts.

- Die Formeln aus der Restriktionsformelmenge RES_v, die alle Haupttestbedingungen und Bestimmungsgleichungen implementieren, werden als Transitionsanschriften zusammengefaßt.

- Die Übergangsprozedur wird durch die PASIPP-spezifischen Prädikate entferne(anzahl,vorkommnis) und einfuege(anzahl,vorkommnis) implementiert[50)]. Hierfür gelten folgende Zuordnungsregeln:

  - Wenn das Element ($mu_{u.i.d}$,$prä_u(m_{s(u.1).d},...,m_{s(1.Ku).d})$) in der Formelmenge $MTAV_{u.v}$ für ein Prädikatssymbol $Prä_u$ aus dem Vorbereich $VB(tr_v)$ der Transaktion tr_v enthalten ist, dann wird dies als Prädikat entferne(mu_u.i.d,prä_u(m_s(u.1).d,...,m_s(u.Ku)).d) notiert.

  - Wenn das Element ($mu_{u.i.d}$,$prä_u(m_{s(u.1).d},...,m_{s(u.Ku).d})$) in der Formelmenge $MTAN_{u.v}$ für ein Prädikatssymbol $Prä_u$ aus dem Nachbereich $NB(tr_v)$ der Transaktion tr_v enthalten ist, dann wird dies als Prädikat einfuege(mu_u.i.d,prä_u(m_s(u.1).d,...,m_s(u.Ku).d)) notiert.

  Die spezielle Konstruktionsweise der beiden PASIPP-Prädikate "entferne" und "einfuege"[51)] sorgt dafür, daß aus der dynamischen Datenbasis der Sektion "Fakten" für jedes Prädikatsvorkommnis entferne(mu_u.i.d,prä_u(m_s(u.1).d,...,m_s(u.Ku).d)) oder einfuege(mu_u.i.d, prä_u(m_s(u.1).d,...,m_s(u.Ku).d)) die Multiplizität des korrespondieren Fakts fakt(mu_u.d, prä_u(m_s(u.1).d,...,m_s(u.Ku).d)) um die entsprechenden Multiplizitäten aus den Prädikaten "entferne" oder "einfuege" vermindert bzw. vergrößert wird. Hierdurch wird die Übergangsoperation der Transaktion tr_v als Transformation der Referenz- in eine Folgefaktenmenge hypothetisch ausgeführt. Daher bilden die beiden "entferne"- und "einfuege"-Prädikate die zentrale Komponente der Transaktionsimplementierung durch PASIPP-Klauseln.

- Die Posttestbedingungen posttest($FAK_{u.+}$) bedeuten für alle Prädikatssymbole $Prä_u$ aus dem Nachbereich $NB(tr_v)$ der Transaktion tr_v, daß die Markenkapazitäten KAP_m derjenigen Stellen s_m, denen die Prädikatssymbole $Prä_u$ mit $bsk(s_m)=Prä_u$ und u=m zugeordnet sind, nach dem hypothetischen Ausführen der Übergangsoperation der Transaktion tr_v eingehalten werden müssen. Die Gesamtheit dieser Posttestbedingungen posttest($FAK_{u.+}$) wird für alle $u \in INB_v$ durch das PASIPP-spezifische Prädikat "kapatest_b" implementiert[52)].

- Die Posttestbedingungen posttest($FAK_{u.+}$) für alle $u \in IVB_v$ entfallen. Da sie als Tautologien formuliert werden, besitzen sie für die Netzimplementierung keine Bedeutung. Sie können das Schaltverhalten des implementierten Synthetischen Netzes nicht beeinflussen.

- Die UNDO-Prozedur des allgemeinen Übergangsschemas entfällt. Denn die Operationsweise des Softwarepakets PASIPP gewährleistet, daß der alte Zustand einer Netzimplementierung[53)] wiederhergestellt wird, wenn die hypothetische Ausführung der Übergangsoperation einer Transaktion infolge Verletzung mindestens einer Posttestbedingung scheitert.

- Das Übergangsprädikat übergang(vb_c($MTAO_v$)) wird im Softwarepaket PASIPP als Transaktionsprädikat fire([tr_v,X_1,...,X_Lv]) implementiert[54]. Dabei ist das Prädikatsargument [...] eine PROLOG-Liste[55]. Sie besteht zunächst aus dem Namen "tr_v" der implementierten Transaktion tr_v. Hinzu kommen alle Variablen X_l mit $l \in \{1,...,L_v\}$ und $L_v \in \mathcal{N}_0$, die in den Argumenten von Formelvorkommnissen aus den Kantengewichten $MTAV_{u.v}$, $MTAI_{u.v}$ und $MTAN_{u.v}$ der Transaktion tr_v für alle Prädikatssymbole $Prä_u$ aus ihrem Operationsbereich OB_v enthalten sind. Falls die hypothetische Ausführung der Übergangsoperation der Transaktion tr_v erfolgreich verläuft, werden die Variablen X_l aus dem Argument des Transaktionsprädikats fire([tr_v,X_1,...,X_Lv]) jeweils durch diejenigen formalen Objekte gebunden, die im Übergangsprädikat übergang(vb_c($MTAO_v$)) durch die Variablenbindungsfunktion vb_c allen Variablen zugeordnet werden. Die per constructionem identischen Variablenbindungen im Übergangs- und im Transaktionsprädikat werden jeweils durch das Unifizierungskonzept der Programmiersprache PROLOG geleistet.
- Alle voranstehend eingeführten PASIPP-Prädikate werden in einer Klausel zusammengefaßt, die für die Transaktion tr_v spezifisch gilt. Es handelt sich um eine HORN-Klausel folgender Gestalt[56]:

  fire([tr_v,X_1,...,X_Lv]) :-

  entferne(mu_u.i.d,prä_u(m_s(u.1).d,...,m_s(u.Ku).d)),

  ... [für alle $u \in IVB_v$ und für alle $d \in DKV_{u.v}$]

  entferne(mu_u.i.d,prä_u(m_s(u.1).d,...,m_s(u.Ku).d)),

  for_z(v.1),

  ...

  for_z(v.Hv),

  einfuege(mu_u.i.d,prä_u(m_s(u.1).d,...,m_s(u.Ku).d)),

  [für alle $u \in INB_v$ und für alle $d \in DKN_{u.v}$]

  einfuege(mu_u.i.d,prä_u(m_s(u.1).d,...,m_s(u.Ku).d)),

  kapatest_b.

r) Einer syntaktischen Konvention der Programmiersprache PROLOG entsprechend werden alle Formeln, die in der Sektion "clauses" Transaktionen, Markenkapazitäten oder Fakten repräsentieren, jeweils durch einen Punkt "." abgeschlossen. Gleiches gilt für das faktenspeichernde Prädikat in der "goal"-Sektion.

s) Die Programmbereiche "domains", "database", "predicates" und "goal" besitzen Turbo-PROLOG-spezifischen Charakter. Für sie existiert im Softwarepaket PASIPP keine unmittelbare Entsprechung[57]. Sie werden hier dennoch aufgeführt. Denn sie gestatten es, die Kombination aus Signaturkonzept und konventioneller Prädikatenlogik 1. Stufe in informationstechnisch präziser Weise zu implementieren. Eine solche sortierte Prädikatenlogik wird von PASIPP zwar mittelbar beherrscht[58], kann aber von diesem Programmpaket nicht unmittelbar repräsentiert werden. Daher hat der Verf. schon an früherer Stelle für eine Reimplementierung von PASIPP in der Programmiersprache Turbo-PROLOG plädiert. Für den Fall, daß diese Reimplementierung tatsächlich erfolgt[59], geben die o.a. Programmbereiche bereits die Grundstruktur des entsprechenden Turbo-PROLOG-Programmpakets vor.

t) Mit der Notation "/* <text> */" werden in ein Turbo-PROLOG-Programm beliebige Kommentare <text> eingefügt, die keine Auswirkung auf Ausführungen des implementierten Programms haben[60]. Sie dienen hier der verdeutlichenden Zuordnung zwischen den Sektionen der Netzlegende und den korrespondierenden Bereichen eines Turbo-PROLOG-Programms.

u) Ein Turbo-PROLOG-Programm, das ein Synthetisches Netz implementiert, nimmt aufgrund aller voranstehend erläuterten Implementierungsvorschriften folgende Gestalt an:

```
/* Netzimplementierung */

/* Marken/Operationssymbole */

domains

        attribut_1 = symbol  / = char  / = string  / = integer / = real
                / = att_j(1)(attribut_q(j(1).1),...,attribut_q(j(1).Kj(1))) [;...]
        ...
        attribut_Q = symbol  / = char / = string  / = integer  / = real
                / = att_j(Q)(attribut_q(j(Q).1),...,attribut_q(j(Q).Kj(Q))) [;...]

        bas_marke = marke_0()

        att_marke_1 = marke_j(1)(attribut_q(j(1).1),...,attribut_q(j(1).Kj(1))) [;...]
        ...
        att_marke_A = marke_j(A)(attribut_q(j(A).1),...,attribut_q(j(A).Kj(A))) [;...]

        str_marke_A+1 = struk_j(A+1)(str_marke_s(j(A+1).1),...,
                str_marke_s(j(A+1).Kj(A+1))) [;...]
        ...
        str_marke_A+B = struk_j(A+B)(str_marke_s(j(A+B).1),...,
                str_marke_s(j(A+B).Kj(A+B))) [;...]

        anzahl = integer
        prädikatsvorkommnis = symbol
        prädikatsname = symbol

/* zeitvariable Faktenmenge */

database

        fakt(anzahl,prädikatsvorkommnis)

/* Stellen/Prädikatssymbole */

predicates

        prä_1(sor_marke_s(1.1),...,sor_marke_s(1.K1))
        ...
        prä_M(sor_marke_s(M.1),...,sor_marke_s(M.KM))

        kapazitaet(prädikatsname,anzahl)
```

clauses

/* Transitionen/Transaktionen */

fire([tr_1,X_1,...,X_L1]) :-
entferne(mu_u.1.d,prä_u(m_s(u.1).d,...,m_s(u.Ku).d)),
... /* für alle $u \in IVB_1$ und für alle $d \in DKV_{u.1}$ */
entferne(mu_u.1.d,prä_u(m_s(u.1).d,...,m_s(u.Ku).d)),
for_z(1.1),
...
for_z(1.H1),
einfuege(mu_u.1.d,prä_u(m_s(u.1).d,...,m_s(u.Ku).d)),
... /* für alle $u \in INB_1$ und für alle $d \in DKN_{u.1}$ */
einfuege(mu_u.1.d,prä_u(m_s(u.1).d,...,m_s(u.Ku).d)),
kapatest_b.

...

fire([tr_N,X_1,...,X_LN]) :-
entferne(mu_u.N.d,prä_u(m_s(u.1).d,...,m_s(u.Ku).d)),
... /* für alle $u \in IVB_N$ und für alle $d \in DKV_{u.N}$ */
entferne(mu_u.N.d,prä_u(m_s(u.1).d,...,m_s(u.Ku).d)),
for_z(N.1),
...
for_z(N.HN),
einfuege(mu_u.N.d,prä_u(m_s(u.1).d,...,m_s(u.Ku).d)),
... /* für alle $u \in INB_N$ und für alle $d \in DKN_{u.N}$ */
einfuege(mu_u.N.d,prä_u(m_s(u.1).d,...,m_s(u.Ku).d)),
kapatest_b.

/* Markenkapazitäten */

kapazitaet(prä_1,kap_1).
...
kapazitaet(prä_M,kap_M).

/* Ausgangsfakten */

fakt(mu_1.d,prä_1(m_s(1).d,...,m_s(K1).d)).
/* für alle $d \in DF_1$ */
...
fakt(mu_U.d,prä_U(m_s(1).d,...,m_s(KU).d)).
/* für alle $d \in DF_U$ */

/* Ausgangsfaktenmenge */

```
goal

        save("fakten.pro").
```

Beispiel einer Netzimplementierung:

Es wird das Beispiel einer Autovermietung aufgegriffen, für das kurz zuvor in Abb. 51 u. 52 ein Synthetischen Netzes in semi-graphischer Darstellungsweise eingeführt wurde. Seine Implementierung durch das Softwarepaket PASIPP nimmt unter den oben eingeführten Anpassungen an die Programmiersprache Turbo-PROLOG folgende Gestalt an:

/* Netzimplementierung */

/* Marken/Operationssymbole */

```
domains

        alter = integer
        baujahr = integer
        kfz_nr = string
        name = string
        strasse = string
        typ = string
        wohnort = string
        adressfeld = adresse(strasse,wohnort)

        kunde = kundenmarke(name,adressfeld,alter)
        auto = automarke(kfz_nr,typ,baujahr)

        anzahl = integer
        prädikatsvorkommnis = symbol
        prädikatsname = symbol
```

/* zeitvariable Faktenmenge */

```
database

        fakt(anzahl,prädikatsvorkommnis)
```

```
/* Stellen/Prädikatssymbole */

predicates

        mietwilliger_kunde(kunde)
        verfuegbares_auto(auto)
        kunde_mit_vertrag(kunde,auto)
        verliehenes_auto(auto)

        kapazitaet(prädikatsname,anzahl)


clauses

/* Transitionen/Transaktionen */

        fire([vertragsabschluss,Alter,Baujahr,Kfz_nr,Name,Strasse,Typ,Wohnort]) :-
                entferne(1,mietwilliger_kunde(kundenmarke(Name,
                        adresse(Strasse,Wohnort),Alter))),
                entferne(1,verfuegbares_auto(automarke(Kfz_nr,Typ,Baujahr))),
                Alter≥21,
                einfuege(1,kunde_mit_vertrag(kundenmarke(Name,
                        adresse(Strasse,Wohnort),Alter),
                        automarke(Kfz_nr,Typ,Baujahr))),
                einfuege(1,verliehenes_auto(automarke(Kfz_nr,Typ,Baujahr))),
                kapatest_b.


/* Markenkapazitäten */

        kapazitaet(mietwilliger_kunde,10).
        kapazitaet(verfuegbares_auto,20).
        kapazitaet(verliehenes_auto,20).
        kapazitaet(kunde_mit_vertrag,20).


/* Ausgangsfakten */

        fakt(1,mietwilliger_kunde(kundenmarke("Maier",
                adresse("Kaiserstr. 10","Karlsruhe"),40))).

        fakt(1,mietwilliger_kunde(kundenmarke("Schmidt",
                adresse("Flughafenstr. 12","München"),19))).

        fakt(1,verfügbares_auto(automarke("MA-K 460","Mazda 323",1986)).

        fakt(1,verfügbares_auto(automarke("MA-K 462","VW Golf",1987)).
```

```
    fakt(1,kunde_mit_vertrag(kundenmarke("Schulte",adresse("Domstr. 4","Köln"),22),
            automarke("MA-K 470","Ford Escort",1983))).

    fakt(1,verliehenes_auto(automarke("MA-K 470","Ford Escort",1983)).


/* Ausgangsfaktenmenge */

goal

    save("fakten.pro").
```

Diese PASIPP- und Turbo-PROLOG-basierte Netzimplementierung ist weniger anschaulich als die semi-graphische Darstellungsweise von Synthetischen Netzen, weil ihr die intuitiv zugängliche Ausdruckskraft der Netzgraphik fehlt. Dafür erweist sich die informationstechnische Netzimplementierung aber auch als kompakter. Die aufwendigen Erläuterungen des Zusammenhangs zwischen impliziter Kurznotation in der Netzgraphik und ihrer vollständigen Explizierung in der Netzlegende erübrigen sich bei der Netzimplementierung. Es besteht also ein Gestaltungsdilemma: Die Anschaulichkeit der semi-graphischen Darstellungsweise von Netzmodellen und die Kompaktheit der informationstechnischen Implementierungen von Netzmodellen lassen sich nicht zugleich erreichen. In dieser Arbeit steht das Interesse im Vordergrund, Netzmodelle erstmals und dabei möglichst verständlich zu entfalten. Daher wird die semi-graphische Darstellungsweise bevorzugt. Falls die vorgeschlagenen Modellierungen jedoch zur Lösung realer Probleme praktisch eingesetzt werden sollten, läßt sich die kompakte Netzimplementierung nicht vermeiden. Nur sie gestattet es, mit den Instrumenten der Automatischen Informationstechnik größere Netzmodelle für realistische Problemformulierungen noch handhaben zu können.

Anmerkungen zum Kapitel:

1) Die Transformation der ursprünglichen Realproblembeschreibung in ein Netzmodell wird im nächsten Kapitel behandelt.

2) Bezugspunkt ist hier der Vergleich mit anderen Netzklassen, insbesondere mit Stelle/Transition- und Kanal/ Instanz-Netzen.

3) PASIPP steht - wie bereits kurz erwähnt - für: Programm zur Analyse und Simulation Prolog-beschrifteter Petri-Netze. Es wurde von OBERWEIS, SCHÖNTHALER, SEIB, LAUSEN und Mitarbeitern an den Universitäten Mannheim und Karlsruhe sowie an der Technischen Hochschule Darmstadt entwickelt. Der Verf. dankt besonders Herrn OBERWEIS für die freundliche Bereitschaft, den Quellcode zur Verfügung zu stellen, und für die Erlaubnis der Programmnutzung. Hierdurch war es möglich, die Programmstruktur einschließlich ihrer Portierungs- und Fortentwicklungsmöglichkeiten eingehender zu analysieren. Einführende Beschreibungen des Programmpakets PASIPP und einfache Anwendungsbeispiele finden sich bei OBERWEIS (1987b), S. 21ff.; OBERWEIS (1988a); OBERWEIS (1988b), S. 299ff.; OBERWEIS (1988c); OBERWEIS (1989a), S. 11ff.; o.V. (1989g), S. 4.
Eine vollständige Dokumentation des Programms PASIPP liegt derzeit noch nicht vor. Allerdings verfügt der Verf. über ein vollständiges Listing des Quellcodes aller Programmodule, auf die er sich im Einzelfall beziehen wird. Da dieser Quellcode geistiges Eigentum der o.a. Autoren darstellt, hat der Verf. aus schutzrechtlichen Gründen darauf verzichtet, dessen Listing in dieser Arbeit wiederzugeben. Er kann jedoch zur Diskussion von Detailfragen beim Verf. eingesehen werden. Darüber hinaus kann er zu Forschungszwecken von Herrn OBERWEIS bezogen werden.
Implementiert wurden die compilierten Versionen 1.0 und 2.1 des Softwarepakets PASIPP sowie die Programmiersprache PROLOG vom Verf. auf ("IBM-kompatiblen") Personalcomputern des Industriestandards unter dem Betriebssystem MS-DOS, Version 3.3, mit einem Hauptspeicherbedarf von mindestens 512 kB.

4) Vgl. dazu die exemplarischen Beschreibungen von PASIPP-Konstrukten in OBERWEIS (1987b), S. 21f.; OBERWEIS (1988a), S. 5ff.; OBERWEIS (1989a), S. 11ff. Eine vollständige Dokumentation des Softwarepakets PASIPP liegt leider zur Zeit noch nicht vor. Ein vollständiges Programmlisting des Quellcodes steht dem Verf. allerdings zur Verfügung. Im Einzelfall wird darauf Bezug genommen, um einzelne PASIPP-Konstrukte oder deren Modifizierungen zu erläutern. Auf einen vollständigen Abdruck des Listings hat der Verf. aus urheberrechtlichen Gründen verzichtet. Hinweise auf die Bezugsmöglichkeit des Softwarepakets finden sich aber z.B. bei OBERWEIS (1987b), S. 21ff., insbesondere S. 23; o.V. (1989g), S. 1ff., insbesondere S. 4 (unten).

5) Dabei handelt es sich um die Dialekte IF-PROLOG, Arity-PROLOG und C-PROLOG.

6) Die Auswahl dieses PROLOG-Dialekts wird nachfolgend motiviert. Allerdings wird damit nicht behauptet, daß die Sprachvariante Turbo-PROLOG bereits alle Implementierungsbedürfnisse in idealer Weise abdeckt. Als pars pro toto wird darauf hingewiesen, daß sie über keine Standardkonstrukte für die Repräsentation von Zeitgrößen besitzt. Für die Implementierung des Kernkonzepts Synthetischer Netze spielt dies keine Rolle, da es ohnehin nicht auf Zeitaspekte eingeht. Später wird das Kernkonzept aber so erweitert, daß es sie explizite Modellierung von zeitbezogenen Einflußgrößen erlaubt. Für die Implementierung dieser Zeitnetze wäre es willkommen, über eine Implementierungssprache mit zeitbezogenen Standardkonstrukten zu verfügen. Einen interessanten Ansatz in dieser Richtung stellt der PROLOG-Dialekt "TASKLOG" dar; vgl. VARNEY (1988), S. 5f. Er erlaubt die Abfrage einer global definierten Systemzeit über eine reservierte Variable "Time (S. 5f.) und die Festlegung maximaler Zeitdauern für die Erfüllung von Aufgaben (S. 6). Leider stand dem Verf. dieser PROLOG-Dialekt jedoch nicht für ein vertieftes Studium zur Verfügung.

7) Vgl. zu den Segmenten von Turbo-PROLOG-Programmen PROLOG (o.J.), S. 20f., 29, 129ff. u. 140ff.; KINNEBROCK (1988), S. 20f., 30f., 87ff. u. 130.
Diese Segmente werden in dieser Arbeit auch als Programmbereiche oder Sektionen von PROLOG-Programmen bezeichnet.

8) Besondere Beachtung verdienen dabei die "domains"-, "predicates"- und "database"-Sektionen von Turbo-PROLOG. Sie eignen sich hervorragend für die Definition der Marken, der Prädikatssymbole bzw. der Faktenmengen von prädikatenlogischen Netzmodellen. Dies wird im folgenden näher dargelegt.

9) Diese Syntaxkonventionen werden von PROLOG (o.J.), S. 196ff., sowie besonders übersichtlich von KINNEBROCK (1988), S. 119ff., zusammengefaßt.

10) Es handelt sich insofern um eine verkürzte Sprechweise, als Synthetische Netze in ihrer programmierten Turbo-PROLOG-Gestalt tatsächlich noch nicht auf einem Automatischen Informationsverarbeitungssystem implementiert sind. Die Programme verkörpern nur potentielle Netzimplementierungen. Da die Realisierung einer potentiellen Netzimplementierung lediglich einen informationstechnischen Routineprozeß darstellt, wird er hier nicht weiter beleuchtet. Statt dessen werden die PROLOG-Programme ohne nähere Differenzierung zwischen ihrer potentiellen und ihrer realisierten Implementierung kurz als Netzimplementierungen bezeichnet.

11) Strenggenommen müßte von implementierungsvorbereiteten Netzen gesprochen werden. Aufgrund der voranstehenden Erläuterungen, insbesondere infolge der Abstraktion vom letzten Schritt der Realisierung potentieller Netzimplementierungen, wird jedoch vereinfacht von implementierten Netzen geredet.

12) Eine Portierung der PROLOG-Dialekte des Softwarepakets PASIPP in Turbo-PROLOG würde insbesondere zusätzliche Typdeklarationen der verwandten Objekte in einer "domains"-Sektion und der benutzen Prädikatssymbole in einer "predicates"-Sektion verlangen. Grundsätzliche Probleme bereitet die Portierung nach Einschätzung des Verf. nicht. Allerdings fehlen in Turbo-PROLOG einige spezielle Metaprädikate, welche die prozedurale Semantik von PASIPP auf der Metaebene der Kontrolle von Inferenzprozeduren beeinflussen (wie z.B. das Prädikat "call"). Ihre Übersetzung in gleichwertige Turbo-PROLOG-Konstrukte ist zwar grundsätzlich möglich, aber nicht trivial. Vgl. zu diesen Defiziten von Turbo-PROLOG bei der Verfügbarkeit von Metaprädikaten FRIEDRICHS,J. (1987), S. 196.

13) Allerdings werden im jeweils betroffenen Einzelfall Modifizierungen von PASIPP besonders herausgestellt, sofern sie über die bloße Reimplementierung in Turbo-PROLOG hinaus wesentliche konzeptionelle Aspekte betreffen.

14) Da der Netzimplementierung ausschließlich die Programmiersprache Turbo-PROLOG zugrundeliegt, wird dieser PROLOG-Dialekt fortan auch einfach nur als "PROLOG" angesprochen.

15) Beispielsweise stellen die "Prädikate" in der "predicates"-Sektion immer Prädikatssymbole dar, im "clauses"-Programmbereich dagegen prädikatenlogische Formeln. Die "Funktionen" und "Konstanten" in der "domains"-Sektion bedeuten Funktions- bzw. Konstantensymbole.

16) Ihre tatsächliche Natur ergibt sich bei der Betrachtung desjenigen Synthetischen Netzes, das der Netzimplementierung durch ein Turbo-PROLOG-Programm zugrundeliegt. Im Synthetischen Netz und seiner semi-graphischen Darstellung sind Prädikatssymbole und prädikatenlogische Formeln, Funktionssymbole und Funktionen sowie Konstantensymbole und Konstanten durch ihre Notationen jeweils wohlunterschieden.

17) Vgl. KINNEBROCK (1988), S. 127 i.V.m. S. 128 (dort werden allerdings Prädikatssymbole verkürzt als Prädikate angesprochen).

18) Vgl. KINNEBROCK (1988), S. 31.

19) Vgl. PROLOG (o.J.), S. 132.

20) Vgl. PROLOG (o.J.), S. 35ff., 130 u. 198; KINNEBROCK (1988), S. 66 u. 128 i.V.m. S. 119ff.

21) Das Formulierungspotential der Turbo-PROLOG-Syntax für symbolische Ausdrücke präzise zu bestimmen, wird hier unterlassen, weil es sich um einen informationstechnischen Aspekt ohne modellierungstechnische Relevanz handelt. Sie kann z.B. PROLOG (o.J.), S. 35ff., 130 u. 198; KINNEBROCK (1988), S. 66 u. 128 i.V.m. S. 119ff., entnommen werden. Grob gesprochen handelt es sich um jede beliebige Kombination aus Buchstaben (des Standardalphabets), Ziffern (des Dezimalsystems) und dem Unterstrich "_", sofern die Kombination durch einen Kleinbuchstaben eingeleitet wird, sowie um Zeichenfolgen aus dem ASCII-256-Standardzeichensatz, sofern sie beidseitig von doppelten Anführungszeichen (") eingeschlossen wird. Hinzu kommen technische Anforderungen an die höchstzulässigen Symbollängen und nähere Spezifikationen der hier als bekannt vorausgesetzten Zeichensätze.

22) Es werden nicht alle Zeichen aus dem ASCII-Standardzeichensatz mit 256 Zeichen zugelassen (ASCII steht für: American Standard Code of Information Interchange), sondern nur die Zeichen mit den Codeziffern 33 bis 126; vgl. PROLOG (o.J.), S. 201f.; KINNEBROCK (1988), S. 155. Kontroll-, Leer- und graphische Sonderzeichen können nicht zum Aufbau von Turbo-PROLOG-Zeichenfolgen verwendet werden. Dies Einschränkung wirkt sich in dieser Arbeit jedoch nicht aus.

23) Vgl. PROLOG (o.J.), S. 199; KINNEBROCK (1988), S. 66.

24) Strenggenommen handelt es sich nur um die Ganzzahlen, die größer als -32769 und kleiner als +32768 sind; vgl. PROLOG (o.J.), S. 200; KINNEBROCK (1988), S. 66. Diese informationstechnischen Beschränkungen zulässiger Ganzzahlen, die aus der Wortbreite der PROLOG-Implementierung folgen, wirken sich in dieser Arbeit nicht restriktiv aus. Daher wird die Menge "INTEGER" vereinfachend auch als Menge "aller" Ganzzahlen bezeichnet.

25) Die allgemeinen Einschränkungen, denen die Verwendung von "reellen" Zahlen in Automatischen Informationsverarbeitungssystemen unterliegt, wurden bereits erörtert. In der Programmiersprache PROLOG wird die Größe der Realzahldarstellung wiederum durch die maximal zulässige Wortbreite begrenzt; Näheres dazu bei PROLOG (o.J.), S. 200.

26) Dabei kann es sich nur um Attributsorten handeln, weil in der Marken-Ontologie Synthetischer Netze alle Markensorten in derivativer Weise als Zielsorten von Erzeugungs- und Strukturierungs-Operationssymbolen definiert sind.

27) Die Gleichsetzung von Operation(ssymbol)en und Funktion(ssymbol)en in Turbo-PROLOG wurde oben angeführt. Die Gleichsetzung von nullstelligen Funktoren "functor()" und Konstanten "functor" findet sich explizit in PROLOG (o.J.), S. 38.

28) Aus der Perspektive des Signaturkonzepts handelt es sich dabei um eine multiple Zielsortendefinition. Die betrachtete Nonstandardsorte wird dabei allerdings nicht mehr originär, sondern derivativ definiert. Sie ist dann die Zielsorte von verschiedenen nullstelligen Operationssymbolen. Zu jedem dieser Operationssymbole gehört in der korrespondierenden Algebra eine Operation, die jeweils das leere Argument λ auf genau eine der Konstante aus der globalen Objektmenge "SYMBOL" für die betrachtete Nonstandardsorte abbilden. Die Objektmenge der Nonstandardsorte wird dann auf die Menge jener Konstanten eingeschränkt, die jeweils die Bilder der Operationsanwendungen auf das leere Argument darstellen. Genau diese eingeschränkte Objektmenge wird in Turbo-PROLOG von vornherein für die "originär" definierte Nonstandardsorte als Konstantenauflistung eingeführt.
Ein Beispiel mag diese Zusammenhänge zwischen Signaturkonzept und Turbo-PROLOG verdeutlichen: Betrachtet wird im Rahmen einer Signatur SIG eine zunächst originär definierte Nonstandardsorte "sort". Zusätzlich wird als Zielsorte der Operationssymbole $Op_1{:} \rightarrow$ sort und $Op_2{:} \rightarrow$ sort derivativ in multipler Weise definiert. Der Sorte "sort" ist in einer zugehörigen SIG-Algebra die globale Objektmenge OB_{sort}=SYMBOL zugeordnet. Den Operationssymbolen entsprechen in der SIG-Algebra die Operationen op_1 und op_2 mit der gemeinsamen Objektmenge OB_{sort}=SYMBOL für ihre Nachbereiche und den Deklarationen: $op_1{:} \rightarrow$ SYMBOL, $op_1()=ob_1$ bzw. $op_2{:} \rightarrow$ SYMBOL, $op_2()=ob_2$. Dies kann verkürzt dargestellt werden als: $ob_1,ob_2{:} \rightarrow$ SYMBOL. Wenn in Turbo-PROLOG die gleiche Nonstandardsorte "sort" originär definiert und dabei von vornherein auf die Objektmenge $OB_{sort}=\{ob_1,ob_2\}$ eingeschränkt werden soll, so wird dies dort notiert als: $sort=ob_1;ob_2$.
Die Abweichung zwischen der originären Sortendefinition in Turbo-PROLOG und der derivativen Zielsortendefinition im Signaturkonzept stellt keine Inkonsistenz dar. Vielmehr handelt es sich um formal leicht differierende Behandlungen der gleichen Sorte "sort". Im Signaturkonzept wird die Sorte "sort" originär mit Hilfe der Objektmenge OB_{sort}=SYMBOL definiert. Alle symbolischen Ausdrücke können grundsätzlich formale Objekte aus der Sorte "sort" darstellen. Die zusätzlich eingeführten nullstelligen Operationssymbole Op_1 und Op_2 sind zwar Konstantensymbole, die für diese Sorte "sort" die formalen Objekte $ob_1=op_1()$ bzw. $ob_2=op_2()$ definieren. Aber es wird hierdurch nicht ausgeschlossen, daß zur Sorte "sort" auch andere symbolische Ausdrücke als formale Objekte gehören (offene Objektmengendefinition). In Turbo-PROLOG wird dagegen die Sorte "sort" originär so definiert, daß sie mit $sort=ob_1;ob_2$ die Objektmenge OB_{sort} von vornherein *genau* auf die Elemente ob_1 und ob_2 festlegt (geschlossene Objektmengendefinition): $OB_{sort}=\{ob_1,ob_2\}$. Falls auch in Turbo-PROLOG die offene Objektmengendefinition erfolgen soll, so müßte dies notiert werden als: sort=SYMBOL;ob_1;ob_2. Dann würden sowohl beliebige symbolische Ausdrücke aus der Objektmenge "SYMBOL" als auch die beiden konkreten formalen Objekte ob_1 und ob_2 als Konstituenten der Objektmenge OB_{sort} zugelassen: OB_{sort}=SYMBOL $\{ob_1)$ $\{ob_2\}$=SYMBOL. Im allgemeinen ist bei praktischen Implementierungen diese offene Objektmengendefinition überhaupt nicht erwünscht. Daher wird in Netzimplementierungen auf der Basis von Turbo-PROLOG als Standardfall die geschlossene Objektmengendefinition $sort=ob_1;ob_2$ mit $OB_{sort}=\{ob_1,ob_2\}$ angenommen.

29) Um an der bereits eingeführten Notation "∅" für Kopien der Basismarke festzuhalten, könnte versucht werden, diese unstrukturierten Markenkopien in der Netzimplementierung als Konstanten der Sorte "char" mit bas_marke='∅' zu notieren. Dabei würde die oben eingeführte Turbo-PROLOG-Notation für Einzelzeichen mit dem Begrenzungszeichen ' benutzt. Leider ist diese Notation jedoch nicht zulässig, weil das Symbol "∅" für die Basismarkenkopie im ASCII-Code nicht als Normalzeichen für die Sorte "char" enthalten ist. (Es besitzt die Codeziffer 237. Dies liegt außerhalb der Normalzeichen-Codeziffern, die von 33 bis 126 reichen.)

30) Die Attributausprägungen "positiv" und "negativ" werden dabei als Ausdrücke aus der globalen Objektmenge "SYMBOL" aufgefaßt. Daher werden sie in den nachfolgenden beiden Gleichungen nicht wie Einzelzeichen oder Zeichenfolgen in Begrenzungszeichen ' bzw. " eingefaßt.

31) Es wird darauf zurückgegriffen, daß alle Operationssymbole aus der algebraisch-prädikatenlogischen Netzdefinition in der Programmiersprache PROLOG als Funktionen behandelt werden.

32) Vgl. dazu die analoge induktive Definition beliebig komplex zusammengesetzter Terme durch Operationssymbole im Signaturkonzept. Die dort benutzten Operationssymbole entsprechen den hier verwendeten Funktionen. Da die Terme sortenbezogen definiert wurden, läßt sich die sortentreue Termsynthese ohne Schwierigkeiten auf die Definition komplexer Sorten analog übertragen.

33) Jede Attribut- oder Markensorte, die in der Legende einer semi-graphischen Netzdarstellung durch Operationssymbole "Att", "Marke" oder "Struk" aus originär definierten Sorten aufgebaut ist, wird in der Netzimplementierung aus jenen Sorten mit den entsprechenden Funktoren "att(...)", "marke(...)" bzw. "struk(...)" abgeleitet.

34) Dabei spielt es keine Rolle, ob die Sorte nur originär oder nur derivativ definiert ist oder beides zugleich gilt.

35) Für originär definierte Nonstandardsorten werden die zugehörigen Objektmengen explizit angegeben, indem die Konstanten aus den Objektmengen aufgezählt werden. Für derivativ definierte Nonstandardsorten werden die

Objektmengen nicht expliziert, sondern sind durch die Objektmengen aller involvierten originären Nonstandard- und Standardsorten und die beteiligten Funktoren implizit definiert.

36) Dies gilt auch für die Objektmengen der originär definierten Standardsorten.

37) In der semi-graphischen Darstellungsweise von Synthetischen Netzen wurde die Vereinfachung eingeführt, die Indizes von Stellen s_m und zugeordneten Prädikatssymbolen $Prä_u$ miteinander zu identifizieren: $bsp(s_m)=Prä_u \rightarrow m=u$.

38) Alle PROLOG-Konstrukte, die keine Variablen darstellen, müssen mit einem Kleinbuchstaben beginnen. Indizierungen durch Subskripte können nicht dargestellt und müssen daher durch Indexsubstitute ersetzt werden. Zwischen Prädikatssymbolen und prädikatenlogischen Formeln wird in allen Dialekten der Programmiersprache PROLOG nicht differenziert; sie werden vereinfacht als Prädikate bezeichnet.

39) Vgl. OBERWEIS (1988a), S. 8. Im Gegensatz zur dynamischen Datenbasis der Sektion "database", deren zugehörige Fakten sich im Zeitablauf durch Schalten von Transitionen verändern können, liegt für die Markenkapazitäten der Stellen eine statische Sammlung von Kapazitätsformeln in der Sektion "clauses" vor. Denn diese Markenkapazitäten gelten zeitinvariant.

40) kap_m ist die PROLOG-spezifische Notation der ganzzahligen Markenkapazitäten KAP_m. Markenkapazitäten KAP_m mit $KAP_m \geq 32768$ können in Turbo-PROLOG-Programmen nicht implementiert werden. Vgl. dazu die Anmerkung zur Einschränkung der Sorte "integer". Daher werden alle Markenkapazitäten, welche die o.a. Ungleichung erfüllen - insbesondere auch die unbeschränkte Kapazität $KAP_m=\omega$ - auf dieselbe maximal darstellbare Kapazität $KAP_m=32767$ abgebildet. Hierin liegt eine mögliche Fehlerquelle für Netzimplementierungen, die in der Syntax der Programmiersprache Turbo-PROLOG begründet ist. Sie wirkt sich allerdings auf diejenigen Synthetische Netze, die in dieser Arbeit für die Modellierung Flexibler Fertigungssysteme gestaltet werden, nicht aus.

41) Die Indizierung des Prädikats "fakt" mit dem Index "r" für den jeweils betrachteten Netzzustand entfällt. Denn der Programmbereich "database" trägt in Turbo-PROLOG-Programmen zur Konstitution einer dynamischen Datenbasis bei, deren Inhalt jeweils zustandsspezifische Gültigkeit besitzt. Der jeweils aktuelle Netzzustand für den Datenbasisinhalt wird nicht durch einen zusätzlichen Zustandsindex "r" angezeigt. Darauf wurde bereits eingangs hingewiesen. Darüber hinaus ist in Turbo-PROLOG kein Mechanismus implementiert, um die Zustandsindizes für unterschiedliche Datenbasisinhalte, die jeweils für verschiedene Netzzustände gelten, zu verwalten. Diese Indexdarstellung und -verwaltung geschieht erst durch das netzspezifische Konzept der Erreichbarkeitsmengen und -graphen. Es ist auch im Softwarepaket PASIPP im Modul BAUM für die Programmversion 1.0 und im Modul TREE für die Programmversion 2.1 enthalten.

42) Vgl. dazu die nachfolgende Erläuterung.

43) Seitens der Programmiersprache PROLOG gehören beliebige formale Objekte zur Sorte "prädikatsvorkommnis", da diese Sorte mit der globalen Sorte "SYMBOL" identifiziert wurde. Die spezielle Konstruktion des Softwarepakets PASIPP - insbesondere seiner Prädikate "entferne" und "einfuege" - und das "save"-Prädikat in der "goal"-Sektion des unten folgenden Turbo-PROLOG-Programms sorgen aber dafür, daß nur die oben festgelegten Formelvorkommnisse formale Objekte aus der Sorte "prädikatsvorkommnis" darstellen können.

44) Daher kann sich das Prädikat "fakt" auf die Formelvorkommnisse "$prä_u$" in seinen Argumenten erstrecken, ohne den Kalkül der Prädikatenlogik 1. Stufe zu verlassen. Denn die Argumente von Prädikaten dürfen sich in diesem Kalkül nicht auf andere Prädikate, wohl aber auf Funktionen beziehen. Diese funktionale Behandlung von formalen Ausdrücken, die strenggenommen den Charakter von Prädikaten tragen, ist der zentrale "Trick" der Programmiersprache PROLOG, die metasprachlichen Fakten auf der objektsprachlichen Programmiersprachenebene ausdrücken zu können.

45) Strenggenommen müssen in PASIPP die faktischen Formeln als Prädikate marke(mu_u.d,prä_u(m_s(u.1).d,..., m_s_(u.Ku).d)) eingegeben werden. Der Prädikatsname "marke" könnte jedoch Verwirrung stiften, weil nicht Marken, sondern Fakten spezifiziert werden. Daher hat der Verf. den Prädikatsnamen "marke" aus dem Programmpaket PASIPP hier durch den Namen "fakt" ersetzt, um für größere Klarheit zu sorgen. Bei einer Turbo_PROLOG-Reimplementierung von PASIPP wäre diese Namenssubstitution zwecks größerer Implementierungstransparenz konsequent auszuführen.

46) Die Extension "pro" besitzt letztlich willkürlichen Charakter. Ihre Bezeichnung verweist hier lediglich auf den Sachverhalt, daß es sich um eine Datenbasis handelt, die im Rahmen der Programmiersprache PROLOG angelegt wird. Die Kleinschreibung des Datenbasisnamens ist zwar aus der perspektive des vorherrschenden Betriebssystems "MS-DOS" für Personalcomputer unüblich. Doch wird sie durch die syntaktische Konvention der Programmiersprache Turbo-PROLOG nahegelegt, alle Ausdrücke, die keine Variablen darstellen, mit einem Kleinbuchstaben zumindest einzuleiten und gewöhnlich auch mit Kleinbuchstaben fortzusetzen. Diese Kleinschreibung bleibt aus der Betriebssystem-Perspektive unschädlich, weil MS-DOS-Kommandos die Unterscheidung zwischen Groß- und Kleinbuchstaben ignorieren.

47) Näheres zur dynamischen Datenbasis von Turbo-PROLOG bei PROLOG (o.J.), S. 140ff.; KINNEBROCK (1988), S. 87ff. Im Softwarepaket PASIPP wird eine analoge dynamische Datenbasis realisiert innerhalb des Moduls HELP1 der PASIPP-Version 1.0 sowie innerhalb der Module I-O-MODU und LADEN der PASIPP-Version 2.1.

48) Hiermit korrespondieren die Ausführungen von Übergangsoperationen jener Transaktionen, die jeweils genau einer Transition mittels Netzbeschriftung zugeordnet sind. Dies wird im folgenden ausführlich erläutert.

49) Vgl. dazu einerseits die zugrundeliegenden Definitionen von Netztopologie, -spezifikationen und -legenden. Vgl. andererseits die spezielle Konstruktionsweise der transaktionsimplementierenden Klauseln des Softwarepaket PASIPP. Sie werden dort vor allem im Modul OPERATIO der PASIPP-Version 1.0 und im Modul OPERAT der PASIPP-Version 2.1 mit der Hilfe von PASIPP-spezifischen Subprädikaten definiert. Eine kompakte und zugleich allgemeingültige Beschreibung dieser Klauseln findet sich bei OBERWEIS (1988a), S. 7ff. Vgl. zu weiteren Erläuterungen der PASIPP-Klauseln für die Implementierung von Transaktionen OBERWEIS (1988b), S. 302f.; OBERWEIS (1989a), S. 14f.

50) In einer PASIPP-Reimplementierung auf Turbo-PROLOG-Basis sind für die Argumente der beiden Prädikate folgende Sorten zu vereinbaren: anzahl=integer und vorkommnis=SYMBOL.

51) Diese Prädikate werden in den Modulen OPERATIO der PASIPP-Version 1.0 und im Modul OPERAT der PASIPP-Version 2.1 als modulspezifische Subprädikate definiert.

52) Auch dieses Prädikat ist in den Modulen OPERATIO der PASIPP-Version 1.0 und im Modul OPERAT der PASIPP-Version 2.1 definiert. Vgl. dazu auch die Erläuterungen bei OBERWEIS (1988a), S. 7f.

53) Dies betrifft den alten Zustand der dynamischen Datenbasis für die aktuelle Faktenmenge.

54) Das Transaktionsprädikat "fire" wird von OBERWEIS (1988a), S. 7, und OBERWEIS (1988b), S. 302, definiert.

55) Vgl. zur Listendarstellung und -verarbeitung in der Programmiersprache Turbo-PROLOG PROLOG (o.J.), S. 45ff.; KINNEBROCK (1988), S. 51ff.

56) Dabei ist zu beachten, daß die Turbo-PROLOG-Syntax eine spezielle Notation für HORN-Klauseln vorschreibt, welche von der prädikatenlogisch üblichen Schreibweise abweicht. Prämisse und Konklusion des Subjugats, das jede HORN-Klausel darstellt, werden in ihrer Anordnung vertauscht. Der prädikatenlogische Operator "→" für Subjugate wird durch die Notation ":-" ersetzt. Falls die Subjugatsprämisse ein Konjugat darstellt, werden dessen Teilformeln aufgelistet. Anstelle des verknüpfenden prädikatenlogischen Konjunktors " " tritt die PROLOG-Notation ",". Jede HORN-Klausel wird durch einen Punkt "." abgeschlossen. Quantoren für die Formelargumente werden grundsätzlich nicht berücksichtigt. (Implizit werden Allquantoren unterstellt.) Daher gilt mit beliebigen, aber atomaren prädikatenlogischen Formeln for_q, $q \in \{1,...,Q+1\}$ und $Q \in \mathcal{N}_+$ für jede HORN-Klausel mit nicht-leerer Prämisse folgende Äquivalenz zwischen prädikatenlogischer und PROLOG-spezifischer Notation:

$$(for_1 \wedge ... \wedge for_Q) \rightarrow for_{Q+1}$$

⇔ for_Q+1 :- for_Q, ..., for_1.

57) Bei der Benutzung des Softwarepakets PASIPP in seinen vorliegenden Versionen 1.0 und 2.1 brauchen nur die drei Unterbereiche des Programmbereichs "clauses" beachtet zu werden. Im Dialog mit PASIPP muß der Benutzer Dateien für die Fakten der Ausgangsmarkierung eines Netzes, die Markenkapazitäten seiner Stellen und die Transaktionen (Transitionen) festlegen.

58) Vgl. dazu die Erläuterung, daß jedes Formelsystem aus einer sortierten Prädikatenlogik im Prinzip als ein logisch gleichwertiges Formelsystem im Rahmen der konventionellen Prädikatenlogik wiedergegeben werden kann. Dies geschieht auch durch das Softwarepaket PASIPP, das auf einer solchen konventionellen, unsortierten Prädikatenlogik beruht. Vgl. darüber hinaus die Anmerkung, daß jede Sorte mit der Hilfe von subjunktiven, aber unsortierten prädikatenlogischen Ausdrücken reformuliert werden kann.

59) Der Verf. hat an anderer Stelle gerechtfertigt, warum er diese technische Reimplementierungsaufgabe in dieser Ausarbeitung nicht mehr angeht.

60) Vgl. PROLOG (o.J.), S. 27; KINNEBROCK (1988), S. 42.

5.1.3.2 Schnittstellen zur Modellkonzeptualisierung

5.1.3.2.1 Allgemeine Konzeptualisierungsvoraussetzungen

Ziel ist es, für die Schnittstelle zwischen Modellkonzeptualisierungen und den daraus abgeleiteten Netzmodellen eine[1)] Methode zu entwickeln, mit deren Hilfe sich unterschiedliche Formen der Konzeptualisierung von Realproblemen[2)] in Synthetische Netze transformieren lassen. Die Transformationsresultate stellen diejenigen Netzmodelle dar, die in dieser Arbeit für Problemmodellierungen intendiert werden. Die spätere Konkretisierung der Transformationsmethode wird zunächst dadurch vorbereitet, daß ihre Anwendungsbedingungen durch Anforderungen an die zugrundeliegenden Modellkonzeptualisierungen offengelegt werden.

Ausgangspunkt ist die natürlichsprachliche Konzeptualisierung eines Realitätsausschnitts, der für die jeweils betrachtete Modellierungsaufgabe relevant ist. Die Umsetzung dieser natürlichsprachlichen Modellbeschreibung[3)] in ein prädikatenlogisches Objektmodell wird hier nicht im Detail behandelt. Sie setzt zwar Erfahrungen mit der Prädikatenlogik 1. Stufe voraus[4)]. Doch bereitet die prädikatenlogische Reformulierung natürlichsprachlicher Sachverhaltsdarstellungen grundsätzlich keine Schwierigkeiten.

Allerdings wird nicht jede beliebige natürlichsprachliche Sachverhaltsbeschreibung zugelassen. Vielmehr muß sie einen Katalog von Anforderungen erfüllen[5)], die anschließend dargelegt werden. Diese Postulate erfüllen eine Demarkationsfunktion. Sie grenzen das Spektrum natürlichsprachlicher Beschreibungen von Modellierungsobjekten so ein, daß sich auf ihre prädikatenlogische Reformulierung die später vorgestellte Transformationsmethode ohne Schwierigkeiten anwenden läßt. Die Anforderungen an die natürlichsprachliche Konzeptualisierung eines Realproblems wirken sich nach Erfahrung des Verf. nicht restriktiv aus[6)]. Bei der exemplarischen Modellierung eines Flexiblen Fertigungssystems, deren Resultate später präsentiert werden, konnten die Konzeptualisierungspostulate stets erfüllt werden.

Anforderungen an die natürlichsprachliche Konzeptualisierung der Modellierungsaufgabe:

a) Das Realproblem, das einer Modellierungsaufgabe zugrundeliegt, wird in systemtheoretischer Weise konzeptualisiert (System-Prämisse). Das natürlichsprachliche Objektmodell bildet daher den problemspezifischen Realitätsausschnitt als ein System ab[7)]. Es wird dadurch zur Repräsentation eines Problems, daß eine nicht-triviale - "problematische" - Diskrepanz besteht zwischen:

- ❑ einem real vorhandenen Ist- oder Ausgangszustand des Systems einerseits sowie
- ❑ mindestens einem erwünschten, aber noch nicht realisiertem Soll- oder Zielzustand desselben Systems andererseits.

Hinzu kommt eine nicht-leere Menge von Aktionen, durch deren Ausführen jeweils ein alter realer in einen neuen realen Systemzustand transformiert werden kann.

b) Entsprechend zur systemtheoretischen Problemkonzeptualisierung läßt sich das gesamte Wissen, das ein Modellierungsträger über das zu modellierende Objekt besitzt, in eine zustands- und ein aktionsbeschreibende Wissenskomponente differenzieren. Die Unterscheidung beider Wissensarten besitzt grundlegende Bedeutung, weil sie auf verschiedene Weisen in ein prädikatenlogisches Objektmodell umgesetzt werden.

c) Das zustandsbeschreibende Wissen wird hier zunächst nur auf den Ausgangszustand des Modellierungsobjekts bezogen. Dies beruht auf zwei Gründen. Erstens gelten die nachfolgenden Ausführungen für die Zielzustände analog, so daß sie nicht ausdrücklich behandelt zu werden brauchen. Zweitens können in Netzmodellen Zielzustände aus der natürlichsprachlichen Objektmodellierung vorerst noch nicht aufgenommen werden. Dies setzt eine pragmatische Erweiterung von Netzmodellen um Zielmarkierungen voraus, die erst später eingeführt wird. Um die nachfolgenden Erläuterungen von vornherein für Ausgangs- und Zielzustände auszulegen, wird allgemein von Modellzuständen gesprochen.

d) Für die Beschreibung von Modellzuständen und zustandsverändernden Aktionen wird ein objektorientierter Gestaltungsansatz gewählt (Objekt-Prämisse). Er knüpft primär an der Abbildung realer Objekte an. Dabei stellt jedes "reale" Objekt bereits das Ergebnis[8)] einer systemtheoretisch ausgerichteten Konzeptualisierung des Realproblems durch einen Modellierungsträger dar[9)]. Reale Objekte werden als Entitäten vorgestellt, die im jeweils modellieren Realitätsausschnitt selbständig existieren[10)]. Zur Objektbeschreibung stehen folgende Ausdrucksmittel bereit:

- ❑ Jedes reale Objekt wird entweder als ein strukturloses Objekt wahrgenommen oder als ein strukturiertes Objekt.
- ❑ Alles, was über ein strukturloses Objekt in einem Zeitpunkt bekannt sein kann, ist dessen aktuelle Existenz.
- ❑ Jedes strukturierte Objekt ist entweder atomar oder zusammengesetzt.
- ❑ Das gesamte Wissen, das über ein atomares strukturiertes Objekt in einem Zeitpunkt vorliegt, ist durch die aktuellen Ausprägungen seiner Eigenschaften (Attribute)[11)] eindeutig bestimmt.
- ❑ Die Eigenschaftsausprägungen sind entweder atomar oder aber aus anderen Eigenschaftsausprägungen in beliebig komplexer Weise zusammengesetzt.
- ❑ Das gesamte Wissen, das über ein zusammengesetztes strukturiertes Objekt in einem Zeitpunkt vorhanden ist, erstreckt sich auf dessen aktuelle Zusammensetzung aus anderen strukturierten Objekten[12)].
- ❑ Zwischen mehreren realen Objekten können Beziehungen (Relationen) beliebiger Art bestehen.
- ❑ Über die Eigenschaften von und die Beziehungen zwischen realen Objekten können beliebig komplexe Feststellungen getroffen werden, solange sich diese Sachverhaltsbeschreibungen auf die:

  - Negation ("nicht"),
  - Konjunktion ("und"),
  - Adjunktion ("oder"),
  - Disjunktion ("entweder ... oder"),
  - Subjunktion ("wenn ..., dann ...") oder
  - Bijunktion ("... genau dann, wenn ...")

  von Eigenschafts- oder Beziehungsfeststellungen zurückführen lassen.
- ❑ Die Gesamtheit aller realen Objekte, die in einem prädikatenlogischen Modell repräsentiert werden können, wird durch eine nicht-leere Menge von Objektklassen strukturiert. Reale Objekte heißen artgleich, wenn sie zur selben Objektklasse gehören.
- ❑ Alles weitere Wissen über einen Modellzustand, das sich auf keine realen Objekte erstreckt, besitzt die Gestalt strukturloser Aussagen. Solche Aussagen besitzen zwar als natürlichsprachliche Sätze jeweils Subjekte und Prädikate. Doch werden diese Satzteile nicht als

selbständig existierende Objekte bzw. als deren Eigenschaften oder Beziehungen konzeptualisiert.

- Durch die Ausführung einer Aktion werden Objekteigenschaften, Objektbeziehungen oder zustandsbeschreibende Aussagen verändert. Triviale Voraussetzung jeder Aktionsausführung ist, daß die jeweils zu verändernden Objekteigenschaften, Objektbeziehungen bzw. zustandsbeschreibenden Aussagen vor der Aktionsausführung auch vorgelegen haben und daß nach der Aktionsausführung kein unzulässiger Modellzustand vorliegt. Weitere, aber nichttriviale Voraussetzungen können hinzukommen. Hierdurch läßt sich festlegen, daß bestimmte Objekteigenschaften, Objektbeziehungen bzw. zustandsbeschreibende Aussagen vor der Aktionsausführung vorliegen müssen, aber durch die Aktionsausführung nicht modifiziert werden.

Der objektorientierte Gestaltungsansatz korrespondiert mit den ontologischen Postulaten, die an früherer Stelle für die "Markenontologie" von Synthetischen Netzen erhoben wurden. Insbesondere wird an die dort entfaltete Persistenzvorstellung angeknüpft, daß die gestalteten Objekte real mit allen ihren Eigenschaften fortbestehen, auch wenn in einem Modellausschnitt Teile dieser Eigenschaften keine Relevanz besitzen.

e) Alle Ausdrücke der Problembeschreibung besitzen einen endlichen Charakter (Finitheits-Prämisse). Sowohl aktuell als auch potentiell unendliche Ausdrücke sind ausgeschlossen. Dies bedeutet u.a., daß die atomaren strukturierten Objekte jeweils nur endlich viele Eigenschaften aufweisen können. Auch die Definitionsbereiche zulässiger Ausprägungen für diese Objekteigenschaften müssen jeweils endliche Mengen darstellen[13)]. Weiterhin werden die Objektklassen als beliebig große, aber stets endliche Mengen angenommen. Es werden nur endlich viele Objektklassen zugelassen. Folglich ist die Anzahl aller Objekte, die in einem Modellzustand dieselbe Eigenschaft besitzen oder in einer bestimmten Beziehung zueinander bestehen, immer endlich[14)]. Schließlich bedeutet die Finitheits-Prämisse, daß endlich viele Objekteigenschaften, -beziehungen und Aussagen ausreichen, um einen Modellzustand zu beschreiben. Ebenso muß eine endliche Anzahl von Aktionen genügen, um alle zulässigen Zustandsveränderungen des Objektmodells zu bewirken[15)].

f) Alle Beschreibungen von Objekteigenschaften und -beziehungen sowie alle zustandsbeschreibenden, aber strukturlosen Aussagen werden stets in gültiger Weise formuliert (Gültigkeits-Prämisse[16)]): Es läßt sich nur die Gültigkeit von eigenschafts-, beziehungs- oder zustandsbeschreibenden Formeln unmittelbar ausdrücken[17)]. Falls die Ungültigkeit solcher Formeln gemeint ist, so wird dies jeweils indirekt als Gültigkeit der kontradiktorischen Formeln dargestellt. Eine solche Transformation ungültiger in gültige Beschreibungsformeln ist aufgrund der voranstehenden Finitheits-Prämisse immer möglich[18)]. Sie wird später konkretisiert. Aufgrund der Gültigkeits-Prämisse nimmt das gesamte Wissen des Modellierungsträgers über einen Modellzustand die Gestalt gültiger Eigenschafts- und Beziehungsfeststellungen oder gültiger (wahrer) Aussagen an.

g) Die Finitheits-Prämisse wird in der Weise verschärft, daß für jede Objekteigenschaft und für jede Objektbeziehung eine maximale Anzahl von Objekten bzw. Objekttupeln bekannt ist, welche die Eigenschaft bzw. Beziehung in einem beliebigen Modellzustand zugleich erfüllen können. Falls diese maximale Objekt- bzw. Objekttupelanzahl nicht explizit spezifiziert ist, wird dafür stets eine beliebig große, aber endliche Zahl implizit unterstellt.

h) In der natürlichsprachlichen Problembeschreibung werden nur generalisierende, aber keine existenzbehauptenden Objektquantifizierungen zugelassen (Quantifizierungs-Prämisse). Es kann also über die Eigenschaften oder Relationen "aller" Objekte aus einer Objektklasse gesprochen werden, nicht aber über die Existenz eines einzelnen *unbestimmten* Objekts. Falls über die Eigenschaften von oder Beziehungen zwischen einzelnen Objekten beschrieben werden sollen, muß stets aus inhaltlich *wohlbestimmte* Objekte Bezug genommen werden[19]. Dies entspricht der Einstellung des mathematisch-logischen Konstruktivismus, einzelne Objekte jeweils konkret anzugeben, anstatt die Existenz nicht näher bestimmter Objekte zu behaupten.

i) Allerdings wird die Quantifizierungs-Prämisse nur als eine schwache Prämisse vorausgesetzt, die verletzt werden darf. Der Verf. teilt zwar die voranstehenden konstruktivistischen Argumente zugunsten der Quantifizierungs-Prämisse. Dennoch läßt er bei der prädikatenlogischen Objektmodellierung grundsätzlich die Verwendung von Existenzquantoren zu. Dadurch wird einerseits dem Umstand Rechnung getragen, daß die konstruktivistische Ablehnung von Existenzquantoren nur die Qualität von Plausibilitätsargumenten besitzt. Sie gilt also nicht stringent für jeden Rezipienten. Andererseits wird durch die Zulässigkeit von Existenzquantoren die hier entfaltete Transformationsmethode so allgemein konzipiert, daß sie den *gesamten* prädikatenlogischen Ausdrucksreichtum zu bewältigen vermag. Falls die schwache Quantifizierungs-Prämisse verletzt wird, muß allerdings zu dem anspruchsvollen Instrument der Formel-Skolemisierung gegriffen werden, um Existenzquantoren formal handhaben zu können[20].

j) Wenn Eigenschaften von oder Beziehungen zwischen realen Objekten ausgedrückt werden sollen, ohne die betroffenen Objekte selbst näher zu bestimmen, dann werden die Objekte durch Variablen vertreten. Jede Variable in einer Sachverhaltsbeschreibung wird global behandelt. Daher können alle Vorkommnisse derselben Variable immer nur die gleichen Objekte ersetzen[21].

k) Die Zusammenfassung von realen Objekten zu Klassen muß vollständig sein. Daher gehört jedes Objekt zu mindestens einer Klasse. Die Disjunktheit der Objektklassen wird jedoch nicht gefordert. Deshalb darf dasselbe Objekt ein Mitglied von mehreren verschiedenen Objektklassen sein[22]. Lediglich die Klasse aller unstrukturierten Objekte muß disjunkt bezüglich aller anderen Objektklassen sein. Dadurch werden unstrukturierte und strukturierte Objekte grundsätzlich voneinander unterschieden. Für alle strukturierten Objekte wird dagegen eine multiple Klassenzugehörigkeit zugelassen. Nur auf diese Weise ist es möglich, daß die Objektklassen in einem hierarchischen Über- und Unterordnungsverhältnis zueinander stehen[23]. Denn ein Objekt, daß zu einer Unterklasse gehört, muß notwendig auch ein Mitglied aller Oberklassen sein, welche die erstgenannte Unterklasse enthalten. Eine solche hierarchische Klassenanordnung hat sich bei der Modellierung realer Objekte vielfach bewährt. Insbesondere stellt sie auch ein herausragendes Merkmal des objektorientierten Gestaltungsansatzes dar, der dieser Arbeit zugrundegelegt wurde. Daher wird dem Verzicht auf disjunkte Objektklassen hier große Bedeutung zugemessen.

l) Die Beschreibung realer Objekte braucht nicht eindeutig zu erfolgen. Statt dessen darf dasselbe reale Objekt auf verschiedene Weisen konzeptualisiert werden[24]. Dann werden die unterschiedlichen Objektbeschreibungen im Objektmodell wie eigenständige Objekte behandelt. Jedes dieser Objekte besitzt eine beschreibungsspezifische interne Struktur. Entsprechend werden die unterschiedlichen Beschreibungen desselben realen Objekts durch verschiedene strukturierte Objekte modelliert. Dadurch läßt sich ein reales Objekt innerhalb desselben Modells je nach Bedarf in unterschiedlichen Erscheinungsweisen erfassen[25].

m) Die beiden Möglichkeiten, dasselbe reale Objekt sowohl in unterschiedlichen Objektklassen einzuordnen als auch verschieden strukturiert zu beschreiben, gestatten einen "Perspektivismus" der Objektmodellierung: Reale Objekte können innerhalb desselben Modells aus unterschiedlichen Konzeptualisierungsperspektiven verschiedenartig wahrgenommen und entsprechend differenziert behandelt werden.

Wenn von der natürlichsprachlichen Konzeptualisierung einer Modellierungsobjekts alle voranstehend erläuterten Anforderungen erfüllt werden, läßt sich dieses natürlichsprachliche Objektmodell in ein prädikatenlogisches Objektmodell umsetzen. Für das prädikatenlogische Objektmodell werden einige zusätzliche Postulate eingeführt[26]. Jedes prädikatenlogische Objektmodell, das auf der Erfüllung der Anforderungen an die natürlichsprachliche Objektmodellierung beruht und alle zusätzlichen Postulate für die prädikatenlogische Objektmodellierung erfüllt, zeichnet sich durch nachfolgend zusammengefaßte Eigenschaften aus.

Charakteristika der prädikatenlogischen Objektmodellierung:

a) Jedes reale Objekt aus der natürlichsprachlichen Modellierung wird im prädikatenlogischen Modell als ein formales Objekt[27] in der Gestalt eines K-Tupels mit $K \in \mathcal{N}_0$ dargestellt. Unstrukturierte (formale) Objekte treten als Nulltupel $() = \varnothing$ auf. Strukturierte (formale) Objekte bilden K-Tupel $(te_1, \ldots, te_K)$ mit $K \in \mathcal{N}_+$. Bei ihren Komponenten handelt es sich um teilevaluierte Terme[28] te_k für alle $k \in \{1, \ldots, K\}$. Diese Terme können Konstanten, Variablen oder Zusammensetzungen aus Konstanten, Variablen und Funktoren (Operatoren[29]) darstellen: $te_k = ob_k$, $te_k = X_k$ bzw. $te_k = fun_j(te_1, \ldots, te_{Kj})$. Alle objektrepräsentierenden K-Tupel sind potentielle Argumente von prädikatenlogischen Formeln.

b) Jeder Klasse realer Objekte entspricht im prädikatenlogischen Objektmodell genau eine Sorte[30]. Da das Modellierungsziel darin besteht, Objekt- in Netzmodelle zu transformieren, werden nicht beliebige Sorten zugelassen. Statt dessen wird von vornherein jede Sorte, die eine Klasse realer Objekte vertritt, als eine sortierte Marke sor_marke_s eingeführt. Die K-Tupel, die als formale Objekte die klassenzugehörigen realen Objekte repräsentieren, sind dann die Kopien der sortierten Marke sor_marke_s.

c) Die Objektmenge OB_s einer Sorte sor_marke_s, die eine Klasse realer Objekte repräsentiert, umfaßt alle formalen Objekte, die reale Objekte aus der sortenzugehörigen Objektklasse darstellen. Wenn keine anderen Festlegungen erfolgen, wird für alle Objektmengen die Voreinstellung $OB_s = SYMBOL$ gewählt. Dadurch wird verhindert, den Formulierungsreichtum für formale Objekte ohne konkrete Veranlassung von vornherein einzuschränken.

d) Die Eigenschaften (Attribute) von strukturierten Objekten können auf zwei alternative Weisen repräsentiert werden. Erstens ist es möglich, sie als Attributsorten $attribut_q$ darzustellen. Zweitens lassen sie sich durch 1-stellige Prädikatssymbole wiedergegeben. Welche dieser beiden Optionen realisiert wird, stellt letztlich eine willkürliche Modellierungsentscheidung dar. Sie kann von Objekteigenschaft zu Objekteigenschaft unterschiedlich ausfallen. Das gilt sogar für verschiedene Eigenschaften desselben Objekts. Auf den hierdurch eröffneten Freiheitsgrad wird später ausführlicher eingegangen. Eigenschaften von unstrukturierten Objekten lassen sich dagegen immer nur durch 1-stellige Prädikatssymbole repräsentieren. Denn die objektrepräsentierenden Nulltupel besitzen keine Komponenten, welche die Eigenschaftsausprägungen darstellen könnten.

e) Zunächst wird der Fall betrachtet, daß eine Objekteigenschaft als eine Attributsorte $attribut_q$ repräsentiert wird. Dann definiert die sortenspezifische Objektmenge OB_q alle formalen Objekte, die zulässige Eigenschaftsausprägungen darstellen. Solange keine anderen Festlegungen erfolgen, werden diejenigen Objektmengen benutzt, die bereits im Kontext des Signaturkonzepts eingeführt wurden[31].

f) Bei der Repräsentation von Objekteigenschaften durch Attributsorten werden Objektklassen und -eigenschaften formal gleich behandelt. Dies entspricht zwar einer Eigentümlichkeit des Konzepts objektorientierter Systemgestaltung, der zufolge kein formaler Unterschied zwischen Objekten und ihren Eigenschaften besteht. Es widerspricht jedoch der ontologischen Differenz zwischen:

- einerseits realen Objekten, die als selbständig existenzfähige Entitäten konzeptualisiert wurden, und
- andererseits Objekteigenschaften, die ohne reale Objekte als Eigenschaftsträger nicht selbständig existieren können.

Auf diese ontologische Unzulänglichkeit des objektorientierten Gestaltungskonzeptes wurde bereits im Zusammenhang mit der Erörterung von Prädikat/Transition-Netzen hingewiesen. Um die daraus folgenden Schwierigkeiten zu vermeiden, wurde eine spezielle "Marken-Ontologie" eingeführt. Im prädikatenlogischen Objektmodell wird sie durch das Postulat berücksichtigt, daß sich die Argumente aller sortierten Prädikatssymbole $Prä_u$ nur auf sortierte Marken, nicht aber auf Attributsorten erstrecken dürfen. Dadurch wird erreicht, daß sich im prädikatenlogischen Objektmodell zustands- oder aktionsbeschreibende Prädikatssymbole $Prä_u$ zunächst immer auf Klassen realer Objekte beziehen: $Prä_u(sor\_marke_{s(u.1)} \ldots sor\_marke_{s(u.K_u)})$. Folglich erstrecken sich die daraus abgeleiteten prädikatenlogischen Formeln $prä_u$ stets auf einzelne reale Objekte: $prä_u(te_1,\ldots,te_{K_u})$ mit $keval_{s(u.k)}(te_k) \in OB_{s(u.k)}$ für alle $k \in \{1,\ldots,K_u\}$. Dadurch wird die unzulängliche Gleichbehandlung von Objekten und deren Eigenschaften ausgeschlossen. Statt dessen nehmen alle Sachverhaltsbeschreibungen im Objektmodell zunächst auf die formalen Repräsentationen derjenigen reale Objekte Bezug, die bei der Konzeptualisierung des modellierten Realitätsausschnitts als selbständig existierende Entitäten wahrgenommen wurden. Erst danach können Objekteigenschaften beschrieben werden. Dabei werden die Sorten aus den Argumenten von Prädikatssymbolen in Subsorten und die Terme in den Argumenten von prädikatenlogischen Formeln in Subterme ausdifferenziert. Die Subsorten repräsentieren dann Eigenschaften und die Subterme Eigenschaftsausprägungen der zuvor erfaßten Objektklassen bzw. Objekte.

g) Objekteigenschaften lassen sich ebenso durch 1-stellige Prädikatssymbole $Prä_u(sor\_marke_{s(u)})$ repräsentieren. Der Name des Prädikatssymbols bezeichnet dann jeweils eine zulässige Eigenschaftsausprägung. Falls für eine Objekteigenschaft insgesamt A unterschiedliche Ausprägungen zulässig sind, werden für die Darstellung dieser Eigenschaft A verschiedene Prädikatssymbole $Prä_{u.a}(sor\_marke_{s(u)})$ mit $a \in \{1,\ldots,A\}$ benötigt, die jeweils über demselben Argument $sor\_marke_{s(u)}$ definiert sind. Daraus abgeleitete prädikatenlogische Formeln $prä_{u.a}(te)$ drücken aus, daß reale Objekte, die durch den Term "te" vertreten werden, die Ausprägung "a" derjenigen Eigenschaft aufweisen, die durch das Prädikatssymbol $Prä_u$ repräsentiert wird[32]. Diese A-fache Darstellung einer Objekteigenschaft bereitet erheblichen Aufwand. Daher wird sie seltener verwendet als die zuvor erläuterten Attributsorten. Dafür entfallen aber auch die voranstehend angesprochenen ontologischen Schwierigkeiten. Denn ein 1-stelliges Prädikatssymbol bezieht sich in seinem Argument von vornherein auf eine sortierte Marke, die ihrerseits eine Klasse realer Objekte vertritt.

h) Alle Beziehungen (mehrstelligen Relationen) zwischen mehreren realen Objekten werden durch mindestens 2-stellige prädikatenlogische Formeln abgebildet. Dabei entspricht die Stelligkeit der Formeln genau der Anzahl der aufeinander bezogenen Objekte. Die Prädikatssymbole, die den Formeln zugrundeliegen, legen durch die sortierten Marken in ihren Argumenten fest, aus welchen Klassen die Objekte jeweils stammen müssen.

i) Jede zustandsbeschreibende, aber strukturlose Aussage wird durch eine 0-stellige prädikatenlogische Formel $prä_u()$ repräsentiert. Ihr liegt jeweils ein 0-stelliges Prädikatssymbol $Prä_u()$ zugrunde. Die "Argumente" von Prädikatssymbolen und Formeln sind zu 0-Tupeln degeneriert[33].

j) Die Negationen sowie die kon-, ad-, dis-, sub- und bijunktiven Verknüpfungen von Eigenschaftsausprägungen, Objektbeziehungen oder Aussagen werden durch die entsprechenden aussagenlogischen Negatoren ($\neg$), Konjunktoren ($\wedge$), Adjunktoren ($\vee$), Disjunktoren ($\underline{\vee}$), Subjunktoren ($\rightarrow$) bzw. Bijunktoren ($\leftrightarrow$) für die jeweils betroffenen prädikatenlogischen Formeln dargestellt.

k) Generalisierende Feststellungen von Eigenschaftsausprägungen oder Objektbeziehungen werden durch Formeln repräsentiert, deren Argumente jeweils mindestens eine Variable enthalten und deren Variablen durch Allquantoren gebunden sind. Existenzbehauptungen über Eigenschaftsausprägungen oder Objektbeziehungen werden entsprechend der Quantifizierungsprämisse zu vermeiden versucht. Falls sie dennoch aufgestellt werden, lassen sie sich zunächst als Formeln darstellen, deren Argumente jeweils mindestens eine Variable umfassen und deren Variablen durch Existenzquantoren gebunden sind. Anschließend werden diese Formeln mittels Skolemisierungs-Operationen in existenzquantorfreie Darstellungsformen transformiert.

l) Offene Formeln, die mindestens eine ungebundene Variable enthalten, werden aus technischen Gründen[34)] grundsätzlich nicht verwendet. Hierin liegt eine effektive Einschränkung gegenüber der prinzipiellen prädikatenlogischen Ausdruckskraft. Bei der Modellierung von Realproblemen ist der Verf. jedoch noch auf keinen Fall gestoßen, in dem sich diese Einschränkung hinderlich ausgewirkt hat[35)].

m) Die prädikatssymbol- und zustandsspezifische Faktenmenge $FAK_{u.r}$ enthält alle Vorkommnisse von atomaren prädikatenlogischen Grundtermformeln[36)] $prä_u(gt_1,...,gt_{Ku})$, die aus einem Prädikatssymbol $Prä_u$ abgeleitet sind und im Zustand "r" des Objektmodells als gültig bekannt sind[37)]. Das gesamte Wissen über Sachverhalte, die in einem Modellzustand "r" als gültig bekannt sind, ist die zustandsspezifische Faktenmenge FAK_r[38)]. Die Interpretation I_r eines prädikatenlogischen Objektmodells drückt dieses Wissen aus, indem sie jedes Prädikatssymbol $Prä_u$ auf dessen Faktenmenge im Modellzustand "r" abbildet.

n) Das faktische Wissen, daß ein atomares Formelvorkommnis $prä_u(gt_1,...,gt_{Ku})$ für das Prädikatssymbol $Prä_u$ im Zustand "r" des Objektmodells gültig ist, läßt sich auf zwei Weisen darstellen. Entweder wird das gültige atomare Formelvorkommnis explizit als Fakt ausgewiesen: $fakt_r(prä_u(gt_1,...,gt_{Ku}))$[39)] ist ein explizit notiertes Fakt. Oder es wird vereinbart, daß alle atomaren Grundtermformeln, die in einem Zustand "r" des Objektmodells explizit aufgeführt werden, implizit als gültig unterstellt werden: $prä_u(gt_1,...,gt_{Ku})$ ist ein implizit notiertes Fakt.

o) Der Verf. bevorzugt im Kontext von Netzmodellen die explizite Faktennotation. Hierfür sprechen drei Gründe. Erstens zeichnet sie sich dadurch aus, daß dem Postulat möglichst weitreichender Konzeptexplizierung entspricht. Zweitens werden Fakten in den später abzuleitenden Netzmodellen ohnehin explizit als faktische Formeln notiert. Drittens erlaubt die explizite Darstellungsweise, kognitionsbedingte Veränderungen von Objektmodellen unmittelbar abzubilden. Denn bei der expliziten Faktennotation ist der Wahrheitswert einer atomaren Grundtermformel $prä_u(gt_1,...,gt_{Ku})$ solange unbekannt, wie nicht die Kenntnis ihrer Gültigkeit durch die faktische Formel $fakt_r(prä_u(gt_1,...,gt_{Ku}))$ explizit dargestellt wird. Daher läßt sich Wissen, das ein Modellierungsträger über die Gültigkeit der Formel $prä_u(gt_1,...,gt_{Ku})$ erworben hat, im Objektmodell durch einen kognitionsbedingten Zustandswechsel ausdrücken[40)]: Es wird von einem Modellzustand, der die faktische Formel $fakt_r(prä_u(gt_1,...,gt_{Ku}))$ noch nicht enthalten hat, zu einem Modellzustand übergegangen, der eben diese Formel umfaßt. Auf diese Weise wird die Menge FAK_r, die alle Fakten des Modellzustands "r" zusammenfaßt, vergrößert[41)]. Kognitionsbedingte Veränderungen eines Objektmodells, die sich auf die Kenntnisse von Formelgültigkeiten erstrecken, spiegeln sich also in entsprechenden Variationen der modellzugehörigen Faktenmengen wieder.

p) In prädikatenlogischen Objektmodellierungen ist es im allgemeinen üblich, Fakten auf implizite Art als Formelvorkommnisse $prä_u(gt_1,...,gt_{Ku})$ zu notieren. Um den Anschluß an diese vertraute Darstellungsweise zu wahren, wird sie für prädikatenlogische Objektmodelle ebenfalls zugelassen. Allerdings muß dann die originäre Gültigkeits-Prämisse durch eine zusätzliche Anforderung verschärft werden. Die originäre Gültigkeits-Prämisse legte lediglich fest, daß sich die Ungültigkeit von Formeln nicht direkt darstellen läßt. Sie traf jedoch keine Festlegung darüber, ob eine atomare Grundtermformel $prä_u(gt_1,...,gt_{Ku})$ aus dem Formelsystem FS als gültig bekannt sein muß. Statt dessen läßt sich mit ihr auch vereinbaren, daß eine atomare Grundtermformel zwar im Formelsystem FS enthalten, aber ihr Wahrheitswert unbekannt ist. Die Gültigkeit einer solchen Formel liegt daher erst dann fest, wenn sie explizit als Fakt $fakt_r(prä_u(gt_1,...,gt_{Ku}))$ notiert wird[42]. Diese Darstellungsweise ist jedoch bei der impliziten Faktennotation unzulässig. Statt dessen soll jedes Vorkommnis einer atomaren Grundtermformel $prä_u(gt_1,...,gt_{Ku})$ implizit als Fakt feststehen. Dies ist nur dann möglich, wenn zusätzlich gefordert wird: Der o.a. Fall, daß der Wahrheitswert einer atomaren Grundtermformel aus dem Formelsystem FS unbekannt ist, wird grundsätzlich ausgeschlossen. Aus der Kombination der originären Gültigkeits-Prämisse, die alle ungültigen Formelvorkommnisse ausgrenzte, mit dem verschärfenden Postulat, das auch alle Formeln mit unbekanntem Wahrheitswert ausschließt, resultiert die verschärfte oder starke Gültigkeits-Prämisse. Ihr zufolge sind *alle* Vorkommnisse von atomaren Grundtermformeln, die im Formelsystem FS enthalten sind, implizit als gültig bekannt[43]. Unter dieser Voraussetzung fallen prädikatenlogische Objektmodelle relativ[44] kompakt aus. Denn es kann auf faktische Formeln $fakt_r(prä_u(gt_1,...,gt_{Ku}))$ verzichtet werden. Der Preis dieser Kompaktifizierung ist allerdings, daß sich kognitionsbedingte Modellveränderungen nicht mehr direkt als Anwachsen der Faktenmenge FAK_r veranschaulichen lassen[45].

q) Falls das Vorkommnis $prä_u(gt_1,...,gt_{Ku})$ einer atomaren Grundtermformel als ungültig bekannt ist, kann dies aufgrund der Gültigkeits-Prämisse[46] nicht unmittelbar ausgedrückt werden. Auch schließt es die Zulässigkeit der impliziten Faktennotation aus, diese Grundtermformel im Objektmodell direkt aufzuführen. Denn sie würde qua Vereinbarung als gültige Formel unterstellt. Statt dessen wird eine kontradiktorische Formel $prä_u^*(te_1,...,te_{Ku})$ eingeführt, die genau dann gültig ist, wenn die Formel $prä_u(te_1,...,te_{Ku})$ ungültig ist: $prä_u^*(te_1,...,te_{Ku}) :\Leftrightarrow \neg prä_u(te_1,..., te_{Ku})$. Daher läßt sich die Ungültigkeit des atomaren Formelvorkommnisses $prä_u(gt_1,...,gt_{Ku})$ im Objektmodell als die Gültigkeit des kontradiktorischen Formelvorkommnisses $prä_u^*(gt_1,...,gt_{Ku})$ ausdrücken[47].

r) Die Wahrheitswerte von geschlossenen quantifizierten Formeln und von atomaren 0-stelligen Formeln werden grundsätzlich in impliziter Weise notiert[48]. Die Gültigkeit von All- und Existenzaussagen bzw. von einzelnen Aussagen wird also dadurch ausgedrückt, daß die betroffenen Formeln im Objektmodell aufgelistet werden. Die Ungültigkeit einer der vorgenannten Formeln wird jeweils als Gültigkeit ihres kontradiktorischen Gegenteils dargestellt.

s) Für jedes Prädikatssymbol ist die endliche Anzahl derjenigen Formelvorkommnisse bekannt, die in einem beliebigen Modellzustand "r" für dieses Prädikatssymbol höchstens gültig sein können. Diese höchstzulässige Anzahl gültiger Formelvorkommnisse stellt eine obere Schranke für die Kardinalität jeder prädikatssymbol- und zustandsspezifischen Faktenmenge $FAK_{u.r}$ dar.

t) Die Interpretation I_0 eines prädikatenlogischen Objektmodells legt für dessen Ausgangszustand "r=0" die Faktenmenge FAK_0 fest. Sie umfaßt alle Vorkommnisse von atomaren Grundtermformeln, die im Ausgangszustand des Objektmodells als gültig bekannt sind. Dabei wird die Gültigkeit der Formelvorkommnisse $prä_u(gt_1,...,gt_{Ku})$ entweder implizit vereinbart oder aber mit Hilfe der faktischen Formeln $fakt_0(prä_u(gt_1,...,gt_{Ku}))$ explizit ausgewiesen. Die Wahrheitswerte aller zusammengesetzten Grundtermformeln ergeben sich daraus indirekt, indem auf die Definitionen der aussagenlogischen Formeloperatoren - der Negatoren und Konnektoren -

zurückgegriffen wird. Darüber hinaus wird allen geschlossenen quantifizierten und allen atomaren 0-stelligen Formeln, die im Objektmodell enthalten sind, jeweils der Wahrheitswert "gültig" zugeordnet. Aufgrund der voranstehenden Festlegungen wird allen Formeln aus dem Formelsystem FS seitens der Interpretation I_0 genau ein Wahrheitswert "gültig" oder "ungültig"[49)] zugewiesen[50)]. Schließlich ordnet die Interpretation I_0 jeder sortierten Marke sor_marke und jeder Attributsorte $attribut_q$ eine sortenspezifische Objektmenge OB_s bzw. OB_q zu[51)]. Diese Objektmengen gelten für alle anderen Modellzustände "r" und zugehörigen Interpretationen I_r unverändert. Sie stellen die Definitionsbereiche der sortenspezifischen Variablen dar.

u) Aufgrund der voranstehenden Erläuterungen gilt die Vereinbarung: In einem prädikatenlogischen Objektmodell ist jede atomare Grundtermformel, jede geschlossene quantifizierte Formel und jede atomare 0-stellige Formel implizit als gültig bekannt, sobald diese Formel im Objektmodell notiert wird. Darüber hinaus kann die Gültigkeit von atomaren Grundtermformeln durch Verwendung der faktischen Formeln $fakt_r(...)$ explizit verdeutlicht werden. Daher stellt das Formelsystem FS eines prädikatenlogischen Objektmodells kein rein syntaktisch definiertes Konstrukt dar. Vielmehr umfaßt es immer schon die - implizite oder explizite - semantische Information darüber, welche Formeln im jeweils betrachteten Modellzustand "r" als gültig bekannt sind. Daher stellt das Objektmodell stets eine Kombination $I_r(FS)$ aus syntaktisch definiertem Formelsystem und semantisch erklärten Formelgültigkeiten dar.

v) Ein prädikatenlogisches Objektmodell liegt in Standardform $I_0(FS)$ vor, wenn gilt:

❑ Das Formelsystem FS ist eine endliche, nicht-leere Menge von prädikatenlogischen Formeln, die implizit konjunktiv verknüpft sind. Die Formelmenge konstituiert den syntaktischen Modellaspekt.

❑ Die Interpretation I_0 ist eine prädikatenlogische Interpretation des Formelsystems FS. Sie stellt den semantischen Modellaspekt dar.

w) Um ein prädikatenlogisches Objektmodell in ein Netzmodell zu transformieren, erweist sich die Standardform $I_0(FS)$ des Objektmodells im allgemeinen als ungeeignet[52)]. Daher werden im anschließenden Kapitel zwei spezielle Darstellungsformen für prädikatenlogische Objektmodelle eingeführt. Es handelt sich um die Klausel- und die Produktionsregelform. Beide erlauben es später, prädikatenlogische Objekt- in Netzmodelle direkt zu transformieren.

Anmerkungen zum Kapitel:

1) Strenggenommen werden zwei Teilmethoden vorgelegt, deren Anwendung noch durch eine Hilfsmethode zur Erzeugung konjunktiver Normalformen vorangestellt wird. Bevor diese methodischen Varianten im einzelnen behandelt werden, wird vereinfacht nur von "einer" Transformationsmethode gesprochen.

2) Mit den Formen der Realproblemkonzeptualisierung sind zunächst die natürlich- und die formalsprachliche Problemkonzeptualisierung gemeint. Die formalsprachliche Konzeptualisierung wird als prädikatenlogisches Objektmodell konkretisiert. Für diese prädikatenlogische Problembeschreibung werden wiederum unterschiedliche Darstellungsformen betrachtet: die Standard-, die Klausel- und die Produktionsregelform.

3) Sie wird auch als natürlichsprachliches Objektmodell bezeichnet. Mit dem "Objekt" ist jeweils das Modellierungsobjekt, d.h. der gesamte konzeptualisierte Realitätsausschnitt gemeint.

4) Daher wurde auf den prädikatenlogischen Grundlagen dieser Arbeit breiter Raum zuteil.

5) Es werden alle natürlichsprachlichen Konzeptualisierungen von Modellierungsobjekten zugelassen, welche die Beschreibungsanforderungen entweder direkt oder aber indirekt erfüllen. Letztes ist der Fall, wenn sich die Konzeptualisierungen in andere natürlichsprachliche Objektbeschreibungen so umwandeln lassen, die ihrerseits die Anforderungen erfüllen.

6) Für andere Modellierungskonzepte sind dem Verf. ähnliche Einschränkungen nicht bekannt. Allerdings sieht er in darin keineswegs einen Hinweis auf deren weniger "restriktive" Auslegung. Vielmehr hegt er den Verdacht, daß dort ähnliche Voraussetzungen hinsichtlich der natürlichsprachlichen Konzeptualisierung von Modellierungsobjekten bestehen, jedoch nicht offengelegt werden. Es liegt allerdings außerhalb des Erkenntnisinteresses dieser Arbeit, jener Vermutung hinsichtlich anderer Modellierungskonzepte näher nachzugehen.

7) Daher werden alle systembezogenen Begriffe analog auf das Objektmodell übertragen. Beispielsweise wird von Modellzuständen gesprochen, wenn die Zustände des objektmodellierenden Systems gemeint sind.

8) Der zugrundeliegende Konzeptualisierungsprozeß der Realitätswahrnehmung wird daher als abgeschlossen vorausgesetzt.

9) Vgl. dazu die früheren Erläuterungen zur Position des kritischen Realismus. Aus dieser Perspektive müßte strenggenommen stets von "realen Objekten in ihrer konzeptualisierten Wahrnehmungsweise" gesprochen werden. Es wurde jedoch bereits an früherer Stelle die Diktionsvereinfachung vereinbart, statt dessen kurz von "realen" Objekten reden zu dürfen.

10) Reale Objekte sind nicht zu verwechseln mit dem jeweils betrachteten Modellierungsobjekt. Erste sind konzeptualisierte Bestandteile des letzten.

11) Der Eigenschaftsbegriff wird im weitesten Sinne verstanden. Er umschließt auch "Befindlichkeiten" von Objekten. Dazu gehören z.B. die Eigenschaften von Werkstücken (Objekten), vor einer Bearbeitungsstation zu warten (wartende Objekte) oder auf einer Station bearbeitet zu werden (sich in Bearbeitung befindende Objekte).

12) Eine Besonderheit stellt das Wissen über die Abwesenheit eines bestimmten anderen Objekts dar. Sie wird hier noch nicht berücksichtigt. Anläßlich der späteren Fallstudie wird sie jedoch als Perspektivenerweiterung eingeführt. Dadurch wird es möglich auszudrücken, daß sich auf dem Objekt "Transportmittel" kein Objekt "Transportobjekt" befindet.

13) Dies erlaubt, nachfolgend die Gültigkeits-Prämisse zu formulieren.

14) Dadurch ist es später möglich, jeder Stelle eines Netzmodells eine endliche Markenkapazität zuzuordnen.

15) Aus beiden letztgenannten Finitheitsaspekten folgt die Endlichkeit der Stellen- und der Transitionenmenge von Netzmodellen.

16) Später wird diese Gültigkeits-Prämisse durch eine zusätzliche Anforderung verschärft. Daher wird die o.a. Gültigkeits-Prämisse zur Abgrenzung von ihrer verschärften Variante auch als originäre oder schwache Gültigkeits-Prämisse bezeichnet.

17) Die Fixierung der Formelgültigkeiten erfolgt durch eine Interpretation I_0 des Formelsystems FS eines Objektmodells. Wesentliches Darstellungsmittel für gültige Formeln sind die Fakten. Daneben enthält die Interpretation I_0 weitere Komponenten, um die Gültigkeit von speziellen Konstrukten - wie z.B. von Allaussagen - auszudrücken.

18) Bei unendlichen Modellierungskonzeptionen könnte dies hingegen nicht mehr garantiert werden. Beispielsweise läßt sich bei unendlich vielen zulässigen Eigenschaftsausprägungen die Ungültigkeit einer bestimmten Eigenschaftsausprägung nicht mehr als Gültigkeit einer kontradiktorischen Ausprägungsfeststellung reformulieren. Beispielsweise ist es möglich, die Ungültigkeit der Objektfarbe "gelb" als Gültigkeit der Adjunktion "blau oder rot" auszudrücken, falls für die Objekteigenschaft "Farbe" nur der endliche Ausprägungsbereich {rot,blau,gelb} zugelas-

sen wird. Bei unendlichem Ausprägungsbereich wäre es dagegen unmöglich, die Ungültigkeit von "gelb" durch irgendeine endliche gültige Adjunktion anderer Farbausprägungen äquivalent zu substituieren.

19) Zur Verdeutlichung mag folgendes Beispiel dienen: Die Feststellung, es "existiere mindestens ein Betriebsmittel mit freier Kapazität", bleibt unbestimmt, solange in einer Modellierung nicht konkret festgelegt wird, *welches* Betriebsmittel hiermit jeweils gemeint ist. Eine wohlbestimmte Modellierung liegt dagegen vor, wenn sie festlegt, daß "das Betriebsmittel mit der identifizierenden Bezeichnung ... über freie Kapazität verfügt". Ausgeschlossen wird hier nur die erste - unbestimmte - Existenzbehauptung, nicht aber die letzte - konkrete - Betriebsmittelangabe.

20) Formeln mit Existenzquantoren werden im allgemein dadurch formal bewältigt, daß sie mit Hilfe der Skolemisierungsoperation in Formeln mit Allquantoren transformiert werden. Auf die Skolemisierung von Formeln mit Existenzquantoren und die semantische Qualität dieser Formeltransformation wurde bereits eingegangen. Dieser Weg wird z.B. beschritten, um die Einschränkung der Programmiersprache PROLOG auf Allquantifizierungen zu rechtfertigen; vgl. CORDES (1988), S. 33; BÖHRINGER (1988), S. 14. Der Übergang zu einer sortierten Prädikatenlogik bereitet keine grundsätzlichen Schwierigkeiten. Denn die Skolemisierungs-Operationen lassen sich auch für diesen Fall definieren; vgl. WALTHER,C. (1987), S. 120ff.
Der Verf. vermeidet durch Erfüllung der Quantifizierungs-Prämisse diese Bewältigungsmöglichkeit von Existenzquantoren, ohne sie jedoch grundsätzlich zu verwerfen. Für diese Vermeidungshaltung sprechen zwei Gründe. Erstens setzt die Formelskolemisierung subtile Kenntnisse der Prädikatenlogik voraus, die der Verf. für den Gebrauch prädikatenlogischer Objektmodelle nicht voraussetzen möchte. Zweitens präferiert er - wie bereits oben dargelegt - die Argumentationsstrategie des mathematisch-logischen Konstruktivismus, grundsätzlich konkret bestimmte Entitäten zu fordern.

21) In formalsprachlichen Kalkülen werden dagegen mitunter Variablen zugelassen, die nur in lokaler Weise gelten. Dann können unterschiedliche Vorkommnisse derselben Variablen durch verschiedene Objekte ersetzt werden, falls die Variablenvorkommnisse nicht zum selben lokalen Bereich gehören. Vgl. dazu die Anmerkungen zur Lokalität der Variablendefinition für die Transitionen von Synthetischen Netzen und für die Programmiersprache PROLOG.

22) Dieser Freiheitsgrad der Konzeptualisierung findet darin seinen Niederschlag, daß die sortenspezifischen Objektmengen überlappen dürfen. Ein formales Objekt aus einer nicht-leeren Schnittmenge repräsentiert dann ein reales Objekt, das in Abhängigkeit von der jeweils gewählten Konzeptualisierungsperspektive verschiedenen Objektklassen zugerechnet wird.

23) Die hierarchische Klassenanordnung wird dadurch erfaßt, daß Subsorten gebildet werden.

24) Diese Definitionsfreiheit wird durch multiple Sortendefinitionen genutzt.

25) Beispielsweise kann ein Fertigungsauftrag in einem Modellmodul, das auf die Auftragsverwaltung zugeschnitten ist, in größerer Detailliertheit abgebildet werden als in einem anderen Modul, das eine Maschine für die Auftragsbearbeitung darstellt.

26) Dabei handelt es sich z.B. um die Voraussetzungen, als formale Konstrukte K-Tupel und Sorten zu benutzen. Ebenso ergibt sich die Gestattung der beiden alternativen Repräsentationsweisen von Objekteigenschaften und ihren Ausprägungen nicht logisch zwingend aus der natürlichsprachlichen Objektmodellierung. Ebenso hätten die prädikatenlogischen Objektmodelle auf genau eine dieser beiden Repräsentationsweisen beschränkt werden können.

27) Sofern aus dem Argumentationskontext ersichtlich ist, daß es sich um die formalen Objekte eines Objektmodells (oder auch aus einem daraus abgeleiteten Netzmodell) handelt, werden sie auch kurz als Objekte bezeichnet.

28) Da ausschließlich teilevaluierte Terme betrachtet werden, werden sie auch einfach nur als Terme angesprochen.

29) Operations- und Funktionssymbole sowie die daraus abgebildeten Operationen bzw. Funktionen und Operatoren bzw. Funktoren werden in dieser Arbeit synonym behandelt. Der Operatorenbegriff wird im Signaturkonzept bevorzugt, der Funktionsbegriff dagegen im prädikatenlogischen Kalkül.

30) Die Umkehrung gilt jedoch nicht. Denn es werden auch Sorten definiert, die keine Objektklassen, sondern Objekteigenschaften darstellen. Die Klasse realer Objekte, die durch eine Sorte repräsentiert werden, heißt auch die sortenzugehörige oder -spezifische Objektklasse.

31) Diese Objektmengen werden auch seitens der hier gewählten Programmiersprache Turbo-PROLOG vorgehalten. Daher erleichtern sie die spätere Implementierung von Netzmodellen, die aus den prädikatenlogischen Objektmodellen abgeleitet werden sollen.

32) Beispielsweise werden für die Darstellung der Objekteigenschaft "Farbe" drei Prädikatssymbole Rot(sor_marke$_{s(u)}$), Blau(sor_marke$_{s(u)}$) und Gelb(sor_marke$_{s(u)}$) benötigt, wenn als zulässige Eigenschaftsausprägungen "rot", "blau" und "gelb" vorgesehen sind. Die Formel blau(X) bedeutet, daß ein beliebiges Objekt, daß für die Variable "X" eingesetzt werden kann, die Eigenschaft "Farbe" in der Ausprägung "blau" besitzt.

33) Diese leeren Argumente können der Einfachheit halber auch fortgelassen werden. In diesem Fall werden die Prädikatssymbole und Formeln als Prä_u bzw. prä_u notiert. Sie stellen dann Aussagensymbole bzw. Aussagen dar. Vgl. dazu den Hinweis, daß die Prädikaten- die Aussagenlogik als Grenzfall umschließt.

34) Netzmodelle werden grundsätzlich von Objektmodellen abgeleitet, deren Formelsysteme jeweils in konjunktiver Normalform vorliegen. In solchen konjunktiven Normalformen werden Allquantoren, die jeweils eine Variable binden, nicht explizit notiert, sondern implizit unterstellt. Folglich gilt für *jedes* Vorkommnis einer Variablen eine implizite Allquantifizierung. (Existenzquantoren wurden zuvor mittels Skolemisierungsoperationen in Allquantoren transformiert.) Daher ist es nicht möglich, in dieser Darstellungsform Variablen zu verwenden, die durch keinen Quantor gebunden sind. Dies entspricht auch der Programmiersprache PROLOG, die später für die Implementierung von Netzmodellen verwendet wird. Dort werden alle Formeln als HORN-Klauseln notiert. In HORN-Klauseln sind aber *alle* Variablen allquantifiziert.

35) Es finden sich auch übereinstimmende Hinweise darauf, daß für alle Variablen in Netzmodellen implizit angenommen wird, sie seien jeweils durch einen Allquantor gebunden; vgl. MURATA,TA. (1988b), S. 485.

36) Die Wahrheitswerte von atomaren Formeln können nur dann bekannt sein, wenn sie keine Variablen enthalten. Daher müssen alle Fakten Grundtermformeln darstellen.

37) Dies hat zur Folge, daß Eigenschaftsausprägungen, die unmittelbar als gültig dargestellt werden sollen, immer als 1-stellige Prädikate formuliert werden müssen. Denn nur solche prädikatenlogischen Formeln können Wahrheitswerte annehmen, nicht aber die Terme aus den Formelargumenten. Daher besteht der o.a. Freiheitsgrad für die Repräsentation von Objekteigenschaften und ihren Ausprägungen nicht mehr, falls die Gültigkeit von Eigenschaftsausprägungen direkt modelliert werden soll. Dagegen läßt sich die Gültigkeit von Eigenschaftsausprägungen weiterhin indirekt abbilden, wenn die Eigenschaftsausprägungen als Terme abgebildet werden und diese Terme in den Argumenten von gültigen Formelvorkommnissen enthalten sind. Dann überträgt sich die Formelgültigkeit mittelbar auch auf die Terme (Eigenschaftsausprägungen) aus den prädikatserfüllenden Argumenten.

38) Die Wahrheitswerte aller zusammengesetzten Formelvorkommnisse ergeben sich mittelbar aus der Kenntnis der Gültigkeit aller atomaren Formelvorkommnisse, die an der Formelzusammensetzung teilnehmen.

39) Wenn mehrere identische Kopien desselben gültigen Vorkommnisses einer atomaren Grundtermformel zugelassen werden, ist eine Notation auf der Basis von Multimengen zu wählen: $\text{fakt}_0(\text{mu}_{u.0},\text{prä}_u(\text{gt}_1,...,\text{gt}_{K_u}))$ drückt aus, daß alle $\text{mu}_{u.0}$ identischen Formelvorkommnisse $\text{prä}_u(\text{gt}_1,...,\text{gt}_{K_u})$ aus der Multimenge $\{(\text{mu}_{u.0},\text{prä}_u(\text{gt}_1,...,\text{gt}_{K_u})):\text{mu}_{u.0}\in \mathcal{N}_+\}$ gültig sind. Diese Multimengen-Notation liegt der unten angeführten allgemeinen Definition von Produktionsregeln zugrunde. Sie wird ebenfalls in der Definition Synthetischer Netze vorausgesetzt. Sie gilt daher auch für die Konstruktion operationaler Objekt- und Netzmodelle. Für die Definition von Klauseln wird sie dagegen nicht benötigt, da in konventionellen Darstellungen von Formelsystemen in Klauselform multiple Prädikatsgültigkeiten nicht bekannt sind. Daher werden in den zugehörigen deklarativen Objekt- und Netzmodellen noch keine Multimengen verwendet. Erst bei der Definition von Produktionsregeln finden sie Berücksichtigung.

40) Vgl. dazu die Ausführungen zu kognitionsbedingten Modellveränderungen.

41) Entsprechend wird die Faktenmenge verringert, wenn Kenntnisse über die Gültigkeit von Formeln untergehen.

42) Dies wurde in der voranstehenden Erläuterung dargelegt.

43) Dies sagt aber nichts über die Wahrheitswerte jener atomaren Grundtermformeln aus, die im Formelsystem FS nicht enthalten sind. Diese Formeln können als gültig bekannt, aber als irrelevant aus dem Formelsystem ausgeschlossen sein. Ebenso kommt in Betracht, daß die Formeln als ungültig bekannt sind oder daß keine Kenntnisse über ihre Wahrheitswerte vorliegen. Es wirt also trotz der starken Gültigkeits-Prämisse keineswegs unterstellt, daß jede atomare Grundtermformel, die im Formelsystem FS nicht vorkommt, ungültig sein müsse.

44) Bezugspunkt sind Objektmodelle mit expliziter Faktennotation.

45) Statt dessen müssen kognitionsbedingte Modellveränderungen indirekt durch Modifizierungen des Formelsystems ausgedrückt werden: Atomare Grundtermformeln dürfen bei impliziter Faktennotation im Formelsystem FS des Objektmodells so lange nicht vorkommen, wie ihr Wahrheitswert unbekannt ist. Sobald sie als gültig oder ungültig bekannt werden, werden die atomaren Formelvorkommnisse $\text{prä}_u(\text{gt}_1,...,\text{gt}_{K_u})$ bzw. deren kontradiktorischen Gegenteile $\text{prä}_u^*(\text{gt}_1,...,\text{gt}_{K_u})$ in das Formelsystem FS aufgenommen und dabei implizit als gültig unterstellt. Durch den Wissenserwerb über die Gültigkeit des Formelvorkommnisses nimmt also der Umfang des Formelsystems FS zu.

Der Verf. steht allerdings dieser indirekten Repräsentation von kognitionsbedingten Modellveränderungen kritisch gegenüber. Denn der *semantische* Modellaspekt, daß die Kenntnisse über Formelgültigkeiten wachsen, wird auf *syntaktische* Weise durch eine Veränderung des Formelsystems FS modelliert. Hierdurch werden kategorial verschiedene semantische und syntaktische Aspekte von prädikatenlogischen Objektmodellen vermengt. Aus diesem Grund bevorzugt der Verf. die o.a. explizite Faktennotation, bei der eine solche Konfusion unterbleibt.

46) Unter den Begriff der Gültigkeits-Prämisse wird fortan sowohl ihre originäre als auch ihre verschärfte Variante subsumiert. Welche der beiden Varianten gemeint ist, hängt davon ab, ob im Argumentationskontext jeweils die explizite bzw. implizite Faktennotation vorausgesetzt wird.

47) Bei expliziter Faktennotation wird diese Gültigkeitsfeststellung als $\text{fakt}_r(\text{prä}_u{}^*(\text{gt}_1,\ldots,\text{gt}_{Ku}))$ notiert. Im Falle der impliziten Faktennotation wird dagegen nur das Formelvorkommnis $\text{prä}_u{}^*(\text{gt}_1,\ldots,\text{gt}_{Ku})$ dargestellt.

48) Eine explizite Darstellung ihrer Gültigkeit, wie es durch den Faktoperator für atomare Grundtermformeln demonstriert wurde, ist in der Prädikatenlogik unbekannt.

49) Der Wahrheitswert "ungültig" kann zusammengesetzten Formeln zukommen.

50) Es könnte zwar eingewandt werden, daß die Wahrheitswerte aller offenen Formeln, die jeweils mindestens eine nicht-quantifizierte Variable enthalten, unter der Interpretation I_0 unbestimmt blieben. Dieser Fall kann jedoch nicht eintreten, weil offene Formeln ausgeschlossen wurden.

51) Die Interpretation würde auch alle Konstanten- und Funktionssymbole auf Konstanten bzw. Funktionen abbilden. Da jedoch von vornherein von teilevaluierten Termen ausgegangen wird, ist diese Abbildungskomponente bereits vorweggenommen.

52) Die Darstellungsform eines Objektmodells wird als geeignet bezeichnet, wenn eine Methode bekannt ist, mit deren Hilfe sich jedes Objektmodell, das diese Darstellungsform erfüllt, in ein Netzmodell transformieren läßt. Für die Standardform $I_0(FS)$ ist eine solche Transformationsmethode nicht bekannt.

5.1.3.2.2 Zwei Konzeptualisierungsformen

Um die Transformation von prädikatenlogischen Objektmodellen in Netzmodelle vorzubereiten, werden die Objektmodelle in zwei speziellen Darstellungsformen aufbereitet. Es handelt sich um deklarative und operationale Objektmodelle. Entsprechend heißen die Netzmodelle, das daraus gewonnen werden können, deklarative bzw. operationale Netzmodelle. Da deklarative und operationale Objektmodellierungen grundverschieden sind, werden die zwei Gestaltungsansätze hinsichtlich ihrer charakteristischen Eigenschaften näher beleuchtet.

Deklarative Objektmodelle

Deklarative Objektmodelle zeichnen sich durch sechs herausragende Merkmale aus:

- Sie beruhen auf der Darstellung prädikatenlogischer Formelsysteme in konjunktiver Normalform.
- Deklarative Objektmodelle erlauben nur die Repräsentation von zustandsbeschreibendem Wissen. Aktionen, deren Ausführungen die jeweils aktuellen Zustände der Objektmodelle verändern würden, lassen sich nicht repräsentieren. Zustandsveränderungen des Objektmodells können allenfalls dadurch bewirkt werden, daß das Wissens des Modellierungsträgers über den jeweils modellierten Realitätsausschnitt zunimmt[1].
- Das Leistungsvermögen deklarativer Objektmodelle konzentriert sich darauf, das implizite Wissen erschließen zu können, das im explizit repräsentierten Wissen eines Objektmodells als die Gesamtheit seiner logischen Konsequenzen enthalten ist[2]. Dabei beziehen sich sowohl das explizite als auch das implizite Wissen auf den jeweils aktuellen Modellzustand "r".
- Grundlegendes Ausdrucksmittel ist ein spezieller Formeltyp: die "Klausel". Es wird daher von Objektmodellen in Klauselform gesprochen. Klauseln sind rein syntaktisch definierte Formeln. Sie lassen sich überwiegend als Subjugate darstellen.
- Fakten erfahren keine besondere Würdigung. Statt dessen werden sie "en passant" als ein spezieller Klauseltyp berücksichtigt, der nicht die vorherrschende Subjugatform annimmt.
- In den Netzmodellen, die sich aus deklarativen Objektmodellen ableiten lassen, spielen Markierungen nur eine technische Rolle. Sie besitzen jedoch keine wohldefinierte Funktion hinsichtlich der Abbildung des jeweils modellierten Realitätsausschnitts.

<u>Erläuterungen und Ergänzungen zu deklarativen Objektmodellen:</u>

a) Das Formelsystem FS eines prädikatenlogischen Objektmodells liegt in seiner konjunktiven Normalform prä_{FS} genau dann vor, wenn sie folgenden Anforderungen genügt:

- Die Formel prä_{FS} ist entweder genau eine Klausel "kla" oder aber ein Konjugat aus endlich vielen, jeweils paarweise verschiedenen Klauseln "kla_p" mit $p \in \{1,...,P\}$, $P \in \mathcal{N}_+$ und $P \geq 2$:

  $\text{prä}_{FS} :\Leftrightarrow \text{kla} \;\underline{\vee}\; ...$

  $\text{prä}_{FS} :\Leftrightarrow \text{kla}_1 \wedge ... \wedge \text{kla}_P$

  Die erste Variante genau einer Klausel wird fortan als Sonderfall der Variante "mehrerer" Klauseln behandelt, für den P=1 gilt. Dann ist jede konjunktive Normalform prä_{FS}[3] ein

"Konjugat" $kla_1 \wedge ... \wedge kla_P$ mit $P \in \mathcal{N}_+$. Für den Sonderfall P=1 degeneriert das "Konjugat" zu der Formel $kla_1 = kla_P = kla$. $KL_{FS} = \{kla_p: p = 1,...,P\}$ ist die nicht-leere Klauselmenge der Formel $prä_{FS}$ in konjunktiver Normalform.

- Jede Klausel kla_p mit $p \in \{1,...,P\}$ ist entweder genau ein Literal lit_p (atomare Klausel) oder aber ein Adjugat aus endlich vielen, jeweils paarweise verschiedenen Literalen $lit_{p.q}$ mit $q \in \{1,...,Q_p\}$, $Q_p \in \mathcal{N}_+$ und $Q_p \geq 2$ (zusammengesetzte Klausel):

  $$kla_p :\Leftrightarrow lit_p \;\underline{\vee}\; ...$$

  $$kla_p :\Leftrightarrow lit_{p.1} \vee ... \vee lit_{p.Qp}$$

  Wiederum lassen sich beide Fälle gemeinsam darstellen als ein "Adjugat" $lit_{p.1} \vee ... \vee lit_{p.Qp}$ mit $Q_p \in \mathcal{N}_+$. Es degeneriert für den Sonderfall Q_p=1 zu der Formel $lit_{p.1} = lit_{p.Qp} = lit_p$. Mit $LIT_p = \{lit_{p.q}: q = 1,...,Q_p\}$ wird die nicht-leere Menge aller Literale notiert, die eine Klausel kla_p konstituieren.

- Jedes Literal $lit_{p.q}$ mit $p \in \{1,...,P\}$ und mit $q \in \{1,...,Q_p\}$ ist entweder das Vorkommnis einer atomaren prädikatenlogischen Formel $prä_{u(p.q)}$ (positives Literal) oder das Vorkommnis ihres Negats (negatives Literal):

  $$lit_{p.q} :\Leftrightarrow prä_{u(p.q)} \;\underline{\vee}\; ...$$

  $$lit_{p.q} :\Leftrightarrow \neg prä_{u(p.q)}$$

- Jedes atomare Formelvorkommnis $prä_{u(p.q)}$ ist eine teilevaluierte Formel, die aus einem K_u-stelligen Prädikatssymbol $Prä_u$ mit $K_u \in \mathcal{N}_0$ abgeleitet ist: $prä_{u(p.q)}(te_1,...,te_{Ku})$[4]. Alle Variablen, die in den teilevaluierten Termen te_k des Formelarguments mit $k \in \{1,...,K_u\}$ enthalten sind, werden implizit durch Allquantoren gebunden[5]. $PRÄ_{FS}$ ist die Menge aller Prädikatssymbole, aus denen jeweils mindestens ein Formelvorkommnis $prä_{u(p.q)}$ eines positiven oder negativen Literals abgeleitet ist.

b) Eine Formel $prä_{FS}$ in konjunktiver Normalform ist in der Regel eine zusammengesetzte Formel. Sie stellt nur dann eine atomare Formel dar, wenn sie aus genau *einer* Klausel besteht (P=1), diese Klausel *atomar* ist (Q_P=1) und durch ein *positives* Literal gebildet wird.

c) Aus der paarweisen Verschiedenheit aller Literale, die in jeweils derselben Klausel vorkommen, folgt: Dasselbe atomare Formelvorkommnis $prä_{u(p.q)}(te_1,...,te_{Ku})$ darf in derselben Klausel höchstens einmal selbst enthalten sein, indem es ein positives Literal $lit_{p.q} :\Leftrightarrow prä_{u(p.q)}(te_1,..., te_{Ku})$ konstituiert, und höchstens einmal in seiner negierten Form vorkommen, indem es ein negatives Literal $lit_{p.q} :\Leftrightarrow \neg prä_{u(p.q)}(te_1,...,te_{Ku})$ aufbaut[6]. Andernfalls wären mehrfache Exemplare desselben (negierten) Formelvorkommnisses redundant, so daß alle - bis auf eines - eliminiert werden könnten. Das gleiche atomare Formelvorkommnis - oder dessen Negat - kann aber redundanzfrei zu unterschiedlichen Klauseln gehören.

d) Die konjunktive Normalform eines Formelsystems enthält weder atomare tautologische Formelvorkommnisse "T" noch atomare kontradiktorische Formelvorkommnisse "⊥". Beide Formelvorkommnisse sind für die prädikatenlogische Objektmodellierung irrelevant[7].

e) Die Unbeachtlichkeit kontradiktorischer atomarer Formelvorkommnisse gilt jedoch nur für die *Darstellung* von prädikatenlogischen Objektmodellen. Dagegen trifft sie nicht mehr auf die *Auswertung* solcher Objektmodelle mit der Hilfe von prädikatenlogischen Inferenztechniken zu. Dies gilt jedenfalls so lange, wie diese Inferenztechniken auf dem Refutationsprinzip beruhen. Dabei wird versucht, eine Behauptung über einen Aspekt des Objektmodells durch Ableiten eines Widerspruchs auf indirekte Art zu beweisen. Ein solcher deduzierter Widerspruch läßt sich stets als kontradiktorisches atomares Formelvorkommnis "$\perp$" ausdrücken. In der Terminologie des Klauselkonzepts handelt es sich um die Ableitung der leeren Klauselmenge $\{\}=\emptyset$. Darauf wurde bereits im Kontext des prädikatenlogischen Resolutionskonzepts hingewiesen. Dort wurde auch vereinbart, in synonymer Weise vom Ableiten der leeren Klausel "$\emptyset$" oder der inkonsistenten Formel "$\emptyset$" zu sprechen. Durch diese Diktion wird es vermieden, im Auswertungszusammenhang die kontradiktorische Formel wiedereinführen zu müssen, die aus dem Darstellungszusammenhang ausgeschlossen worden war. Falls eine genauere Differenzierung erforderlich ist, werden die inkonsistente Formel und die leere Klausel als atypische Klauseln bezeichnet. Alle anderen Klauseln heißen dagegen typische Klauseln[8)].

f) Jede Formel prä in konjunktiver Normalform läßt sich in syntaktisch abweichender, aber semantisch äquivalenter[9)] Weise als eine endliche Klauselmenge KL notieren. Hierfür gilt unter Einschluß der atypischen Klausel:

$$\text{prä} :\Leftrightarrow \qquad \Leftrightarrow \qquad KL = \{\} = \emptyset$$

$$\text{prä} :\Leftrightarrow \text{kla} \qquad \Leftrightarrow \qquad KL = \{\text{kla}\}$$

$$\text{prä} :\Leftrightarrow \text{kla}_1 \wedge \ldots \wedge \text{kla}_P \qquad \Leftrightarrow \qquad KL = \{\text{kla}_p\text{: } p=1,\ldots,P\}$$

g) Eine Klausel heißt subjunktiv, wenn sie aus mindestens einem negativen und mindestens einem positiven Literal besteht. Eine solche Klausel läßt sich als ein Subjugat darstellen. Dabei werden Literale der Klausel unter Ausnutzung des Kommutativ- und Assoziativgesetzes der Adjunktion[10)] so umgruppiert, daß zunächst alle negativen Literale $\text{lit}_{p.q}:\Leftrightarrow\neg\text{prä}_{u(p.q)}$ und danach alle positiven Literale $\text{lit}_{p.q}:\Leftrightarrow\text{prä}_{u(p.q)}$ aufgelistet werden. Die Anzahl der negativen Literale wird mit $Q_{N.p}$ ($Q_{N.p}\in\mathcal{N}_+$) notiert, die Anzahl der positiven Literale mit $Q_{P.p}$ ($Q_{P.p}\in\mathcal{N}_+$). Für eine subjunktive Klausel kla_p aus Q_p Literalen stellt die Folge $1,\ldots,Q_{N.p},Q_{N.p}+1,\ldots,Q_{N.p}+Q_{P.p}$ mit $Q_{N.p}+Q_{P.p}=Q_p$ eine Reindexierung aller Literale $\text{lit}_{p.q}$ aus der Literalemenge LIT_p der Klausel kla_p dar. Dabei werden zunächst die Indizes $1,\ldots,Q_{N.p}$ aller negativen Literale und danach die Indizes $Q_{N.p}+1,\ldots,Q_p$ aller positiven Literale aufgeführt. Unter Ausnutzung der DE MORGAN-Regel für die Negation von Adjugaten[11)] sowie der definitorischen Rückführung von Subjugaten auf Adjugate[12)] gilt dann:

$$\begin{aligned}
\text{kla}_p \quad &:\Leftrightarrow \quad \text{lit}_{p.1} \vee \ldots \vee \text{lit}_{p.Qp}\\
&\Leftrightarrow \quad (\neg\text{prä}_{u(p.q(1))}) \vee \ldots \vee (\neg\text{prä}_{u(p.q(QN.p))}) \vee \text{prä}_{u(p.q(QN.p+1))} \vee \ldots \vee \text{prä}_{u(p.q(Qp))}\\
&\Leftrightarrow \quad (\neg(\text{prä}_{u(p.q(1))} \wedge \ldots \wedge \text{prä}_{u(p.q(QN.p))})) \vee (\text{prä}_{u(p.q(QN.p+1))} \vee \ldots \vee \text{prä}_{u(p.q(Qp))})\\
&\Leftrightarrow \quad (\text{prä}_{u(p.q(1))} \wedge \ldots \wedge \text{prä}_{u(p.q(QN.p))})) \rightarrow (\text{prä}_{u(p.q(QN.p+1))} \vee \ldots \vee \text{prä}_{u(p.q(Qp))})
\end{aligned}$$

i) Eine subjunktive Klausel zeichnet sich durch zwei Charakteristika aus: Erstens enthält sie nur noch atomare Formelvorkommnisse $\text{prä}_{u(p.q)}$, nicht mehr aber deren Negate[13)]. Zweitens sind alle atomaren Formelvorkommnisse im Antezedens des Subjugats konjunktiv und alle atomaren Formelvorkommnisse in der Konklusion des Subjugats adjunktiv verknüpft, sofern das Antezedens bzw. die Konklusion aus mehreren atomaren Formelvorkommnissen bestehen. Die ehemals negierten Literale $\text{lit}_{p.q(d)}$, die nun im Antezedens als atomare Formelvorkommnisse $\text{prä}_{u(p.q(d))}$ mit $d\in\{1,\ldots,Q_{N.p}\}$ enthalten sind, heißen auch Antezedensformeln[14)]. Entsprechend werden die nicht-negierten Literale $\text{lit}_{p.q(d)}$, die nun in der Konklusion als atomare Formelvorkommnisse $\text{prä}_{u(p.q(d))}$ mit $d\in\{Q_{N.p}+1,\ldots,Q_p\}$ vorkommen, als Konklusionsformeln bezeichnet[15)]. Besondere Bedeutung wird später der Sachverhalt erlangen, daß die Konklusionsformeln für $Q_{P.p}\geq 2$ ein Adjugat bilden. Dies wird vereinfacht[16)] als adjunktive Konklusionslogik von subjunktiven Klauseln angesprochen.

j) Falls ein prädikatenlogisches Objektmodell $I_0(FS)$ noch nicht in Klauselform vorliegt, kann sein Formelsystem FS durch nachfolgend beschriebene Transformationsmethode in seine "äquivalente"[17] Klauselform überführt werden. Die Interpretation I_0 für den Ausgangszustand des Objektmodells wird von der Transformation des Formelsystems FS nicht betroffen, da es sich um eine rein syntaktisch definierte Formeltransformation handelt. Die invariante semantische Interpretation I_0 des Formelsystems FS bleibt daher im folgenden unbeachtet. Das Ergebnis des Transformationsprozesses ist genau eine - im Regelfall komplex zusammengesetzte - Formel prä_{FS} in konjunktiver Normalform. Sie entspricht dem ursprünglichen Formelsystem FS des Objektmodells.

k) Das prädikatenlogische Formelsystem FS, das den syntaktischen Aspekt eines beliebigen prädikatenlogischen Objektmodells in Standardform konstituiert, ist entweder eine einzelne, beliebig komplex zusammengesetzte prädikatenlogische Formel oder eine endliche Sammlung mehrerer solcher Formeln. Dem zweiten Fall liegt eine implizit konjunktive Verknüpfung aller Mitglieder der Formelsammlung zugrunde[18]. Dieses prädikatenlogische Formelsystem FS wird durch folgende Transformation in seine konjunktive Normalform prä_{FS} überführt[19]:

- Falls das Formelsystem FS aus mehreren Formeln prä_h[20] mit $h \in \{1,...,H\}$, $H \in \mathcal{N}_+$ und $H \geq 2$ besteht, so wird deren implizit konjunktive Verknüpfung als äquivalentes[21] Konjugat prä_{FS}^*[22] aller systemzugehörigen Formeln expliziert:

  $$FS = \{\text{prä}_h\text{: } h=1,...,H \wedge H \geq 2\} \quad \Leftrightarrow \quad \text{prä}_{FS}^* :\Leftrightarrow \text{prä}_1 \wedge ... \wedge \text{prä}_H$$

  Andernfalls wird das einelementige Formelsystem $FS=\{\text{prä}_1\}$ direkt als die Formel $\text{prä}_{FS}^*:\Leftrightarrow\text{prä}_1$ notiert[23].

- Alle Sub-, Bi- und Disjugate, die in der Formel prä_{FS} vorkommen, werden in äquivalente Ad- und Konjugate transformiert. Dazu dienen folgende Äquivalenzbeziehungen:

  $$\begin{array}{lll} \text{prä}_1 \rightarrow \text{prä}_2 & \Leftrightarrow & (\neg\text{prä}_1) \vee \text{prä}_2 \\ \text{prä}_1 \leftrightarrow \text{prä}_2 & \Leftrightarrow & (\text{prä}_1 \rightarrow \text{prä}_2) \wedge (\text{prä}_2 \rightarrow \text{prä}_1) \\ & \Leftrightarrow & ((\neg\text{prä}_1) \vee \text{prä}_2) \wedge ((\neg\text{prä}_2) \vee \text{prä}_1) \\ \text{prä}_1 \underline{\vee} \text{prä}_2 & \Leftrightarrow & (\text{prä}_1 \wedge (\neg\text{prä}_2)) \vee ((\neg\text{prä}_1) \wedge \text{prä}_2) \end{array}$$

- Alle Negate von zusammengesetzten Formeln werden in äquivalente zusammengesetzte Formeln transformiert, in denen nur noch die Negate von atomaren Formeln vorkommen:

  $$\begin{array}{lll} \neg(\text{prä}_1 \wedge \text{prä}_2) & \Leftrightarrow & (\neg\text{prä}_1) \vee (\neg\text{prä}_2) \\ \neg(\text{prä}_1 \vee \text{prä}_2) & \Leftrightarrow & (\neg\text{prä}_1) \wedge (\neg\text{prä}_2) \\ \neg(\neg\text{prä}) & \Leftrightarrow & \text{prä} \\ \neg(\forall(X)\text{: prä}(X)) & \Leftrightarrow & (\exists(X)\text{: }\neg\text{prä}(X)) \\ \neg(\exists(X)\text{: prä}(X)) & \Leftrightarrow & (\forall(X)\text{: }\neg\text{prä}(X)) \end{array}$$

- Alle Formeln, die mindestens einen Existenzquantor enthalten, werden durch Skolemisierungsoperationen[24] in Formeln transformiert, die keine Existenz-, sondern nur noch Allquantoren umfassen. Diese Skolemisierungsoperationen interessieren hier nicht im Detail, weil aufgrund der Quantifizierungs-Prämisse ihre Verwendung von vornherein vermieden wird[25].

- In allen Formeln, in denen Allquantoren vorkommen, werden die Allquantoren an den Formelbeginn gerückt. Beispielsweise gilt:

$$\forall(X_1)\text{: prä}_1(X_1) \wedge (\forall(X_2)\text{: prä}_2(X_1,X_2))$$
$$\Leftrightarrow \quad (\forall(X_1)\ \forall(X_2)\text{: prä}_1(X_1) \wedge \text{prä}_2(X_1,X_2))$$

- Alle Allquantoren, die Formeln einleiten, werden aus der expliziten Formelnotation eliminiert und fortan für alle Variablen in den Formelargumenten implizit unterstellt. Das voranstehende Beispiel wird etwa notiert als: $\text{prä}_1(X_1) \wedge \text{prä}_2(X_1,X_2)$).

- Alle 0-stelligen Formeln prä() werden durch 1-stellige Formeln prä(∅) ersetzt[26], für die jeweils gilt: Die Substitutformel prä(∅) ist das Vorkommnis eines Prädikatssymbols Prä(bas_marke), dessen Argument durch die strukturlose Basismarke "bas_marke" gebildet wird. Der Term im Argument der 1-stelligen Substitutformel prä(∅) ist immer die Konstante "∅", d.h. die Kopie der Basismarke[27].

- Alle Adjugate, die noch mindestens eine Komponente enthalten, die keine atomare Formel oder deren Negat ist, werden durch Auflösen von Konjugaten und von Adjugateinklammerungen in äquivalente Formeln transformiert, deren Adjugate nur noch aus atomaren Formeln oder deren Negaten bestehen. Beispielsweise gilt:

$$(\text{prä}_1 \wedge \text{prä}_2) \vee \text{prä}_3 \quad \Leftrightarrow \quad (\text{prä}_1 \vee \text{prä}_2) \wedge (\text{prä}_1 \vee \text{prä}_3)$$
$$\text{prä}_1 \vee (\text{prä}_2 \wedge \text{prä}_3) \quad \Leftrightarrow \quad (\text{prä}_1 \vee \text{prä}_2) \wedge (\text{prä}_1 \vee \text{prä}_3)$$
$$(\text{prä}_1 \vee (\neg\text{prä}_2)) \vee \text{prä}_3 \quad \Leftrightarrow \quad \text{prä}_1 \vee (\neg\text{prä}_2) \vee \text{prä}_3$$

l) Nach den voranstehenden Transformationen wird das ursprüngliche Formelsystem FS durch genau eine prädikatenlogische Formel prä_{FS} wiedergegeben, die vier Bedingungen erfüllt: Erstens erscheint die Formel prä_{FS} in konjunktiver Normalform. Zweitens enthält die Formel keine Existenzquantoren (skolemisierte Normalform). Drittens werden alle Variablen, die in der Formel vorkommen, implizit durch Allquantoren gebunden. Viertens nimmt an der Formelzusammensetzung aus atomaren Formeln keine 0-stellige atomare Formel - also keine aussagenlogische Formel - teil. Falls das Formelsystem FS eines Objektmodells in Standardform von vornherein diesen Anforderungen gerecht wird, braucht keine der o.a. Transformationen angewendet zu werden.

m) Aus der konjunktiven Normalform prä_{FS} des Formelsystems FS läßt sich die Klauselmenge KL_{FS} des deklarativen Objektmodells unmittelbar ablesen. Dazu brauchen lediglich die Äquivalenzen aus Erläuterung f) angewendet zu werden[28].

n) Da die Interpretation I_0 eines prädikatenlogischen Objektmodells in Standardform $I_0(FS)$ von den voranstehenden Transformationsmethode nicht berührt wird, gilt für das transformierte Objektmodell in Klauselform: $I_0(\text{prä}_{FS})$ oder $I_0(KL_{FS})$. Beide Varianten des deklarativen Objektmodells unterscheiden sich lediglich in syntaktischer Hinsicht dadurch, ob seine Klauselgesamtheit als Konjugat prä_{FS} oder als Menge KL_{FS} notiert wird.

o) Ein deklaratives Objektmodell, das in der Gestalt $I_0(\text{prä}_{FS})$ notiert wird, heißt ein Objektmodell in konjunktiver Normalform. Da die Formel prä_{FS} per constructionem aus Klauseln besteht, liegt das Objektmodell zugleich in einer Klauselform i.w.S. vor. Die Darstellung des Objektmodells in der Variante $I_0(KL_{FS})$ wird dagegen als seine Klauselform i.e.S. bezeichnet. Falls keine nähere Festlegung erfolgen soll, ob ein deklaratives Objektmodell in syntaktischer Hinsicht durch eine Formel prä_{FS} in konjunktiver Normalform oder durch eine Klauselmenge KL_{FS} ausgedrückt ist, wird von einem Objektmodell in Klauselform gesprochen.

p) Die Fakten eines deklarativen Objektmodells in Standardform $I_0(FS)$ stellen gültige atomare Grundtermformeln dar. Nach der Modelltransformation in seine Klauselform $I_0(KL_{FS})$ liegt jedes Fakt als eine gültige atomare Klausel vor, die aus genau einem positiven Literal besteht und deren Argument keine Variablen enthält. Solche Klauseln werden auch als faktische Klauseln bezeichnet. Wie bei der Standardform können sie sowohl in expliziter als auch in impliziter Weise notiert werden. Im ersten Fall nehmen sie die Gestalt einer faktischen Formel $fakt_0(prä_u(gt_1,...,gt_{Ku}))$ an. Im zweiten Fall handelt es sich dagegen um atomare Grundtermformeln $prä_u(gt_1,...,gt_{Ku})$, deren Gültigkeit implizit vereinbart ist. In beiden Fällen sind die faktischen Klauseln über die gesamte Klauselmenge des Objektmodells verstreut. Bei ihrer impliziten Darstellungsweise fallen diese Klauseln oftmals nicht auf den ersten Blick ins Auge. Dagegen werden faktische Klauseln durch ihre explizite Notation $fakt_0(prä_u(gt_1,...,gt_{Ku}))$ deutlich von allen anderen Klauseln abgehoben. Daher präferiert der Verf. die zweite Darstellungsart.

Operationale Objektmodelle:

Operationale Objektmodelle unterscheiden sich von ihren zuvor behandelten deklarativen Pendants durch sechs komplementäre Merkmale:

- Operationale Objektmodelle bestehen aus Formelsystemen, die keiner der üblichen prädikatenlogischen Normalformen[29)] entsprechen.
- Objektmodelle in Produktionsregelform gestatten es, sowohl zustands- als auch aktionsbeschreibendes Wissen zu repräsentieren. Die Darstellung von Aktionen, deren Ausführungen die Modellzustände verändern, bildet sogar das Schwergewicht der operationalen Wissensformulierung[30)]. Daneben können wie bei deklarativen Objektmodellen kognitionsbedingte Modellveränderungen hinzukommen.
- Das Leistungsvermögen operationaler Objektmodelle konzentriert sich darauf, das Potential aller zulässigen Zustände des Objektmodells erforschen zu können, die durch zustandsverändernde Aktionsausführungen aus einem vorgegebenen Ausgangszustand "r=0" hervorgebracht werden können. Das implizite Wissen eines operationalen Objektmodells besteht in dem Potential aller zulässigen Modellzustände, die sich aus dem zunächst bekannten Ausgangszustand durch Operationsausführungen im Objektmodell ableiten lassen.
- Grundlegendes Ausdrucksmittel sind Produktionsregeln. Es wird daher von einer Produktionsregelform der Objektmodelle gesprochen. Produktionsregeln stellen Formeln dar, die eine syntaktisch definierte Subjugatform mit einem semantisch erklärten Bezug auf Faktenmengen kombinieren. Der semantische Aspekt von Produktionsregeln konzentriert sich auf die Veränderung von Faktenmengen.
- Fakten erhalten einen gleichrangigen Status neben den Produktionsregeln. Beide verhalten sich zueinander komplementär: Die Gesamtheit aller Fakten repräsentiert das vollständige explizite Wissen über einen Modellzustand. Die Menge aller Produktionsregeln drückt dagegen das gesamte explizite Wissen über mögliche Veränderungen von Modellzuständen aus.

- In den Netzmodellen, die sich aus operationalen Objektmodellen ableiten lassen, besitzen Markierungen eine wohldefinierte Funktion. Sie dienen dazu, das explizite Wissen über den jeweils aktuellen Zustand des prädikatenlogischen Objektmodells zu repräsentieren[31)]. Dieses zustandsspezifische Wissen wird in prädikatenlogischen Objektmodellen durch eine ebenso zustandsspezifische Faktenmenge ausgedrückt. Daher lassen sich die Markierungen als Repräsentationen von zustandsbeschreibenden Faktenmengen auffassen[32)].

Erläuterungen und Ergänzungen zu operationalen Objektmodellen:

a) In einem operationalen Objektmodell wird das Formelsystem FS, das alle prädikatenlogischen Formeln für die Modellierung des betrachteten Realitätsausschnitts umfaßt, in zwei Teilsysteme ausdifferenziert: Die Produktionsregelmenge PR_{FS} einerseits und die Faktenmenge $FAK_{FS.0}$ für den Ausgangszustand "r=0" des Objektmodells andererseits. In beiden Mengen gelten die Produktionsregeln bzw. Fakten implizit als konjunktiv verknüpft[33)].

b) Die Faktenmenge $FAK_{FS.0}$ enthält alle atomaren Formelvorkommnisse, die im Ausgangszustand "r=0" des Objektmodells gültig sind. Daher drückt die Faktenmenge bereits einen wesentlichen Aspekt der Interpretation I_0 des Formelsystems FS aus. Die übrigen Aspekte der Interpretation I_0 - wie z.B. die Festlegung der sortenspezifischen Objektmengen[34)] - werden in der Rumpfinterpretation I_0^* zusammengefaßt. Somit kann jedes prädikatenlogische Objektmodell $I_0(FS)$ als ein operationales Objektmodell in Produktionsregelform $(I_o^*(PR_{FS}),FAK_{FS.0})$ dargestellt werden. Wenn von der Rumpfinterpretation I_0^* abgesehen wird[35)], so wird das verbleibende Paar $(PR_{FS},FAK_{FS.0})$ aus einer modellspezifischen Produktionsregel- und Faktenmenge als ein Produktionsregelsystem bezeichnet.

c) Für die Umsetzung der Standardform $I_0(FS)$ eines Objektmodells in dessen Produktionsregelform $(I_o^*(PR_{FS}),FAK_{FS.0})$ kann keine so übersichtliche Transformationsmethode angegeben werden, wie es oben für die Herstellung der Klauselform möglich war. Der Verf. verzichtet darauf, eine entsprechend kompliziertere Transformationsmethode systematisch auszuformulieren. Statt dessen wird nur festgelegt, welche Form die Produktionsregeln und Fakten in den Mengen PR_{FS} bzw. $FAK_{FS.0}$ eines operationalen Objektmodells annehmen müssen. In späteren Beispielen werden Objektmodelle von vornherein so konzipiert, daß sich ihre natürlichsprachlichen Umschreibungen direkt als Fakten oder Produktionsregeln formal darstellen lassen[36)].

d) Fakten nehmen in Produktionsregelsystemen die gleiche Gestalt wie in prädikatenlogischen Objektmodellen in Standard- oder Klauselform an. Sie stellen jeweils gültige Vorkommnisse von atomaren Grundtermformeln dar. Im Gegensatz zu deklarativen Objektmodellen sind die Fakten eines operationalen Objektmodells jedoch nicht über das gesamte Formelsystem verstreut, sondern in der separaten Faktenmenge $FAK_{FS.0}$ deutlich ausgewiesen. Daher spielt hier die Verwendung der expliziten Notationsweise eine weniger wichtige Rolle als bei deklarativen Objektmodellen. Trotzdem bevorzugt der Verf. auch in operationalen Objektmodellen die Verwendung der faktischen Formeln $fakt_0(prä_u(gt_1,...,gt_{Ku}))$. Hierfür sprechen die gleichen Argumente, die bereits oben zugunsten der expliziten Faktennotation angeführt wurden.

e) Produktionsregeln[37)] beschreiben reale Aktionen, die im jeweils modellierten Realitätsausschnitt ausgeführt werden können. Innerhalb eines Objektmodells besitzen sie den Charakter von komplexen Operationen. Die Ausführung einer solchen Operation bewirkt, daß ein alter Modellzustand (Referenzzustand) "r" in einen neuen Modellzustand (Folgezustand) "f" transformiert wird. Die Wirkung einer Regelanwendung besteht also in einer Zustandstransformation. Allerdings läßt sich diese Regelwirkung nicht immer erzielen. Vielmehr handelt es sich bei Produktionsregeln im Regelfall[38)] um bedingte Operationen[39)]. Sie können nur dann ausgeführt werden, wenn eine regelspezifische Voraussetzung erfüllt ist. Diese Regelvoraussetzung muß im

Referenzzustand eines Objektmodells erfüllt sein, damit die Produktionsregel überhaupt angewendet werden darf. Die Regelvoraussetzung läßt sich wie in einem deklarativen Objektmodell durch eine prädikatenlogische Formel ausdrücken. Daher vereinigen Produktionsregeln zwei unterschiedliche Qualitäten in sich[40]: Hinsichtlich ihrer Regelwirkung stellen sie operationale Konstrukte dar. Bezüglich ihrer Regelvoraussetzung besitzen sie dagegen einen deklarativen Charakter.

f) Für Produktionsregeln wird eine Notation benutzt, die einerseits möglichst allgemein formuliert ist, um verschiedenartige Regelkonkretisierungen zu überdecken. Andererseits wird die Regelformulierung speziell auf den früher entfalteten Kalkül der sortierten Prädikatenlogik 1. Stufe und das bereits entfaltete Konzept Synthetischer Netze abgestimmt. Dabei wird eine Darstellungsform gewählt, die für Produktionsregeln unüblich ist[41]. Dennoch wird sie hier verwendet, weil sie sich durch zwei Vorzüge auszeichnet. Erstens fällt die Regelformulierung allgemeiner aus als die sonst vorherrschenden Regeldefinitionen. Zweitens läßt sie die ambivalente, sowohl deklarative als auch operationale Charakteristik von Produktionsregeln besonders deutlich erkennen[42].

g) Eine Produktionsregel ist in syntaktischer Hinsicht ein Subjugat. Sein Antezedens drückt ausschließlich die deklarativ formulierte Regelvoraussetzung aus[43]. Seine Konklusion spezifiziert dagegen in operationaler Weise die vollständige Regelwirkung[44]. Der semantische Gehalt von Produktionsregeln wird dadurch hervorgehoben, daß sich Regelvoraussetzung und -wirkung auf Faktenmengen erstrecken. Ein besonderer "Trick" ermöglicht es, auf die Faktenmengen durch rein syntaktisch definierte Formeln Bezug zu nehmen. Auf diese Weise werden semantische Aspekte des prädikatenlogischen Objektmodells mit rein syntaktischen Mitteln behandelt. Dies entspricht genau der früher erörterten Vorgehensweise, die Gültigkeit atomarer Formeln $prä_u(gt_1,...,gt_{Ku})$ durch faktische Formeln $fakt_r(prä_u(gt_1,...,gt_{Ku}))$ auszudrücken, die ebenfalls nur syntaktisch definierte Konstrukte darstellen.

h) Eine Produktionsregel ist jede prädikatenlogische Formel, die folgende Formelstruktur erfüllt:

- Die Produktionsregel pr_y mit $y \in \mathcal{N}_+$ ist ein Subjugat aus Regelvoraussetzung con_y[45] und Regelwirkung act_y[46]:

  $$pr_y \quad :\Leftrightarrow \quad (con_y \rightarrow act_y)$$

- Die Regelvoraussetzung con_y ist entweder die tautologische Formel "T" oder eine einzelne nicht-tautologische Voraussetzungsformel con_y oder ein Konjugat aus nicht-tautologischen Voraussetzungsformeln $con_{y.v}$ mit $v \in \{1,...,V_y\}$, $V_y \in \mathcal{N}_+$ und $V_y \geq 2$. Für $V_y \in \mathcal{N}_+$ wird vereinfacht von einem "Konjugat" $con_{y.1} \wedge ... \wedge con_{y.Vy}$ gesprochen, das im Sonderfall $V_y=1$ zu genau einer Voraussetzungsformel $con_{y.1} = con_{y.Vy} = con_y$ degeneriert. Wenn $V_y \in \mathcal{N}_0$ angibt, aus wie vielen Teilformeln $con_{y.v}$ die Regelvoraussetzung con_y besteht, dann gilt:

  $$con_y \quad :\Leftrightarrow \quad \begin{cases} T; & \text{für } V_y=0 \\ con_y; & \text{für } V_y=1 \\ con_{y.1} \wedge ... \wedge con_{y.Vy}; & \text{für } V_y \geq 2 \end{cases}$$

  $CON_y = \{T\}$ für $V_y=0$ und $CON_y = \{con_{y.v}: v=1,...,V_y\}$ für $V_y \in \mathcal{N}_+$ ist die Voraussetzungsmenge der Produktionsregel pr_y. Die Regelvoraussetzung spezifiziert diejenige Bedingung, die erfüllt sein muß, damit die Produktionsregel pr_y ausgeführt werden kann[47].

- Jede nicht-tautologische Voraussetzungsformel[48] $con_{y.v}$ mit $v \in \{1,...,V_y\}$ und $V_y \in \mathcal{N}_+$ ist entweder eine faktenbezogene Inklusionsformel $ink_{y.v}$ oder aber eine algebraische Bedingungsformel $bed_{y.v}$:

  $con_{y.v} :\Leftrightarrow ink_{y.v} \;\underline{\vee}\; con_{y.v} :\Leftrightarrow bed_{y.v}$

- Jede faktenbezogene Inklusionsformel $ink_{y.v}$ drückt aus, daß eine Multimenge $MTAI_{u(y.v).n(y)}$ aus teilevaluierten atomaren Formelvorkommnissen $prä_{u(y.v)}(te_{1.d},...,te_{Ku.d})$[49] mit $d \in \mathcal{N}_+$ in der Faktenmenge $FAK_{u(y.v).r}$ eines formelspezifischen[50] Prädikatssymbols $Prä_{u(y.v)}(sor\_marke_{s(u.1)} ... sor\_marke_{s(u.Ku)})$ enthalten ist, falls alle Variablen in den Argumenten der Formelvorkommnisse aus der Multimenge durch eine Variablenbindungsfunktion vb_c auf variablenfreie Grundterme[51] abgebildet worden sind[52]:

  $ink_{y.v} :\Leftrightarrow vb_c(MTAI_{u(y.v).n(y)}) \leq FAK_{u(y.v).r}$

- Jede algebraische Bedingungsformel $bed_{y.v}$ ist eine beliebige atomare K-stellige algebraische Formel $for_{z(y.v)}(te_1,...,te_K)$ mit $K \in \mathcal{N}_+$, sofern sie folgenden Einschränkungen genügt: Ihre teilevaluierten Terme te_k mit $k \in \{1,...,K\}$ können nur entweder Konstanten oder Variablen sein (flache Formelargumente). Für jede Variable aus dem Formelargument muß gelten[53]: Sie ist mindestens einmal in einem Argument derjenigen Formelvorkommnisse $prä_{u(y.v)}(te_{1.d},...,te_{Ku.d})$ enthalten, die zu einer beliebigen Multimenge $MTAI_{u(y.v).n(y)}$ für eine Inklusionsformel $ink_{y.v}$ aus der Regelvoraussetzung con_y gehören. Oder[54] die Variable kommt mindestens einmal in einem Argument derjenigen Formelvorkommnisse $prä_{u(y.w)}(te_{1.d},...,te_{Ku.d})$ vor, die zu einer beliebigen Multimenge $MTAV_{u(y.w).n(y)}$ für eine Reduktionsformel $red_{y.w}$ aus der Regelwirkung act_y gehören. Die Reduktionsformeln $red_{y.w}$ werden in Kürze näher beschrieben. Über alle Variablen aus dem Argument einer Bedingungsformel erstreckt sich jeweils ein implizit vereinbarter Allquantor[55].

- Die Regelwirkung act_y ist entweder eine einzelne Wirkungsformel act_y oder ein Konjugat aus Wirkungsformeln $act_{y.w}$ mit $w \in \{1,...,W_y\}$, $W_y \in \mathcal{N}_+$ und $W_y \geq 2$. Für $W_y \in \mathcal{N}_+$ wird vereinfacht von einem "Konjugat" $act_{y.1} \wedge ... \wedge act_{y.W_y}$ gesprochen, das im Sonderfall $W_y=1$ zu genau einer Wirkungsformel $act_{y.1} = act_{y.W_y} = act_y$ degeneriert. Wenn $W_y \in \mathcal{N}_+$ die Anzahl aller Wirkungsformeln $act_{y.w}$ in der Regelwirkung act_y angibt, dann gilt:

  $$act_{y.w} :\Leftrightarrow \begin{cases} act_y; & \text{für } W_y=1 \\ act_{y.1} \wedge ... \wedge act_{y.W_y}; & \text{für } W_y \geq 2 \end{cases}$$

  $ACT_y = \{act_{y.v}: v=1,...,W_y\}$ ist die Wirkungsmenge der Produktionsregel pr_y.

- Jede Wirkungsformel $act_{y.w}$ mit $w \in \{1,...,W_y\}$ ist entweder eine faktenbezogene Modifikationsformel $mod_{y.w}$ oder eine algebraische Bestimmungsformel $bes_{y.w}$:

  $act_{y.w} :\Leftrightarrow mod_{y.w} \;\underline{\vee}\; act_{y.w} :\Leftrightarrow bes_{y.w}$

- Jede faktenbezogene Modifikationsformel $mod_{y.w}$ ist entweder eine Reduktionsformel $red_{y.w}$ oder aber eine Expansionsformel $exp_{y.w}$:

  $mod_{y.w} :\Leftrightarrow red_{y.w} \;\underline{\vee}\; mod_{y.w} :\Leftrightarrow exp_{y.w}$

- Jede Reduktionsformel $red_{y.w}$ beschreibt, daß eine Multimenge $MTAV_{u(y.w).n(y)}$ aus teilevaluierten atomaren Formelvorkommnissen $prä_{u(y.w)}(te_{1.d},...,te_{Ku.d})$ mit $d \in \mathcal{N}_+$ aus der Faktenmenge $FAK_{u(y.w).r}$ eines formelspezifischen[56] Prädikatssymbols $Prä_{u(y.v)}(sor\_marke_{s(u.1)} ... sor\_marke_{s(u.Ku)})$[57] entfernt wird, falls alle Variablen in den Argumenten der Formelvorkommnisse aus der Multimenge mittels einer Variablenbindungsfunktion vb_c[58] durch Grundterme ersetzt worden sind. Dabei wird die alte Faktenmenge $FAK_{u(y.w).r}$ für den Referenzzustand "r" des Objektmodells in die neue Faktenmenge $FAK_{u(y.w).f}$ für den Folgezustand "f" des Objektmodells überführt:

$$red_{y.w} \;:\Leftrightarrow\; FAK_{u(y.w).f} := FAK_{u(y.w).r} - vb_c(MTAV_{u(y.w).n(y)})$$

- Jede Expansionsformel $exp_{y.w}$ drückt aus, daß eine Multimenge $MTAN_{u(y.w).n(y)}$ aus teilevaluierten atomaren Formelvorkommnissen $prä_{u(y.w)}(te_{1.d},...,te_{Ku.d})$ mit $d \in \mathcal{N}_+$ zu der Faktenmenge $FAK_{u(y.w).r}$ eines formelspezifischen[59] Prädikatssymbols $Prä_{u(y.v)}(sor\_marke_{s(u.1)} ... sor\_marke_{s(u.Ku)})$[60] hinzugefügt wird, sofern alle Variablen in den Argumenten der Formelvorkommnisse aus der Multimenge durch eine Variablenbindungsfunktion vb_c auf Grundterme abgebildet worden sind. Dabei wird die alte Faktenmenge $FAK_{u(y.w).r}$ für den Referenzzustand "r" des Objektmodells in die neue Faktenmenge $FAK_{u(y.w).f}$ für den Folgezustand "f" des Objektmodells transformiert:

$$exp_{y.w} \;:\Leftrightarrow\; FAK_{u(y.w).f} := FAK_{u(y.w).r} + vb_c(MTAN_{u(y.w).n(y)})$$

- Jede algebraische Bestimmungsformel $bes_{y.w}$ weist einer Variablen X_k unter einer Variablenbindungsfunktion vb_c einen eindeutigen Wert zu, der das Bild einer formelspezifischen, H_k-stelligen Bestimmungsfunktion $bestimme_k(X_1,...,X_{Hk})$ mit $H_k \in \mathcal{N}_+$ und $k \in \{1,...,H_k\}$ ist. Die Variable X_k muß mindestens einmal in einem Argument derjenigen Formelvorkommnisse $prä_{u(y.v)}(te_{1.d},...,te_{Ku.d})$ vorkommen, die zu einer beliebigen Multimenge $MTAN_{u(y.w).n(y)}$ aus einer Expansionsformel $exp_{y.w}$ der Regelwirkung gehören[61]. Die Variablen X_h mit $h \in \{1,...,H_k\}$ müssen jeweils mindestens einmal in einem Argument derjenigen Formelvorkommnisse enthalten sein, die zu einer beliebigen Multimenge aus einer Inklusionsformel $ink_{y.v}$ der Regelvoraussetzung oder aus einer Reduktionsformel $red_{y.w}$ der Regelwirkung gehören. Für alle Variablen, die in den Argumenten der Formelvorkommnisse aus allen Multimengen $MTAN_{u(y.w).n(y)}$ enthalten sind, ist jeweils genau eine solche Bestimmungsformel definiert[62]. Aufgrund der voranstehenden Festlegungen erweist sich jede Bestimmungsformel als eine atomare algebraische Formel, für die mit $k \in \{1,...,H_k\}$ gilt:

$$bed_{y.v}(X_k;X_1,...,X_{Hk}) \;:\Leftrightarrow\; vb_c(X_k) = vb_c(bestimme_k(X_1,...,X_{Hk}))$$

  Über alle Variablen aus dem Argument einer Bestimmungsformel erstreckt sich jeweils ein implizit vereinbarter Allquantor.

i) Produktionsregeln heißen typisch, wenn ihre Regelvoraussetzung wegen $V_y \in \mathcal{N}_+$ aus mindestens einer nicht-tautologischen Voraussetzungsformel $con_{y.v}$ besteht. Andernfalls handelt es sich für $V_y=0$ um eine atypische Produktionsregel. Da die Regelvoraussetzung einer atypischen Produktionsregel die tautologische Formel darstellt, ist ihre Regelvoraussetzung immer erfüllt. Infolgedessen kann sich die Regelvoraussetzung einer atypischen Produktionsregel auf die Regelausführung nicht restriktiv auswirken. In diesem Sinne wird auch von einer voraussetzungsfreien oder unbedingten Produktionsregel gesprochen[63].

j) Die algebraischen Bedingungsformeln stellen zumeist eine derjenigen algebraischen Vergleichsformeln dar, die im Zusammenhang mit den SIG-Spezifikationen eingeführt wurden. Allgemein handelt es sich um beliebige atomare algebraische Formeln, die entweder aus der standardisierten Formelmenge $FOR_{standard}$ stammen oder aber im Rahmen der Produktionsregeldefinition speziell eingeführt werden. Die algebraischen Bestimmungsformeln sind dagegen kraft ihrer Definition von vornherein auf Formeln mit Gleichungscharakter festgelegt.

k) Die algebraischen Bedingungs- und Bestimmungsformeln mögen zwar auf den ersten Blick als artifizielle ad hoc-Konstrukte erscheinen. Doch sie spielen bei der Formulierung von Prädikat/Transition-Netzen und ähnlichen Höheren Netzen als Transitionsbeschriftungen oftmals eine große Rolle[64]. Darüber hinaus führen sie nach der Transformation von Objektmodellen in Netzmodelle zu Haupttestbedingungen[65] und Bestimmungsgleichungen[66], die das allgemeine Übergangsschema ÜS von Synthetischen Netzen für einzelne Transaktionen konkretisieren. Schließlich lassen die algebraischen Formeln in prädikatenlogischen Objektmodellen das algebraische Formulierungspotential deutlicher erkennen, als es aus den allgemeinen prädikatenlogischen Formeln $prä_u$ ersichtlich ist. Aufgrund aller voranstehenden Argumente werden die Bedingungs- und Bestimmungsformeln in dieser Arbeit explizit in die Formulierung von Produktionsregeln einbezogen. Dagegen sind sie in der Literatur, die sich speziell mit Produktionsregelsystemen beschäftigt, unüblich[67].

l) Es wird vorausgesetzt, daß sich die Menge aller Prädikatssymbole $Prä_u$, die in der Regelvoraussetzung con_y einer Produktionsregel pr_y vorkommen, und die Menge aller Prädikatssymbole $Prä_u$, die in der Regelwirkung act_y derselben Produktionsregel pr_y enthalten sind, zueinander disjunkt verhalten. Hierdurch erhalten Produktionsregeln eine besonders klare Struktur, in der die Aspekte der Voraussetzung und der Wirkung von Regelausführungen vollständig voneinander getrennt werden. Zugleich wird sichergestellt, daß alle Formelvorkommnisse, die in den Multimengen der Regelvoraussetzung enthalten sind, zwar gültig sein müssen, um eine Produktionsregel überhaupt ausführen zu können. Von der tatsächlichen Regelausführung werden diese Formelvorkommnisse in ihrer Gültigkeit jedoch nicht beeinträchtigt[68]. Darüber hinaus erfüllt die Disjunktheitsforderung eine teleologische Funktion. Sie stellt sicher, daß die Integritätsbedingung IB_I von Synthetischen Netzen eingehalten wird, wenn operationale Objekt- in entsprechende Netzmodelle transformiert werden. Denn erst die o.a. Disjunktheitsvoraussetzung garantiert, daß der Informationsbereich jeder regelabbildenden Transition kein gemeinsames Prädikatssymbol mit deren Wirkungsbereich besitzt.

m) $PR_{FS} = \{pr_y\text{: } y=1,...,Y\}$ ist die Menge aller Produktionsregeln pr_y, die mit $y \in \{1,...,Y\}$ und $Y \in \mathcal{N}_+$ für ein Objektmodell in Produktionsregelform definiert sind. Sie können entweder aus einem zugrundeliegenden Objektmodell in Standardform mit dem Formelsystem FS in informaler Weise abgeleitet sein. Oder es wurde von vornherein ein operationales Objektmodell mit der Hilfe von Produktionsregeln formuliert.

n) Wenn von einem operationalen Objektmodell in Produktionsregelform ausgegangen wurde, dann läßt sich ihm nachträglich ein Formelsystem FS zuordnen. Es besteht zunächst aus der Menge aller faktenbezogenen[69] oder algebraischen[70] Formeln, die an der Konstitution aller Produktionsregeln pr_y aus der Regelmenge PR_{FS} teilnehmen. Hinzu kommen alle explizit oder implizit repräsentierten Fakten, die in der Faktenmenge $FAK_{FS.0}$ enthalten sind.

o) Zwischen Produktionsregeln in der hier entfalteten Form und den transitionsspezifischen Schaltregeln von Synthetischen Netzen besteht eine tiefe strukturelle Verwandtschaft[71]. Die Schaltregel einer Transition läßt sich als eine bedingte Anweisung zur Veränderung der Netzmarkierung auffassen[72]: Die Anweisungsbedingung erstreckt sich auf die Aktivierungsbedingung der Transition. Der Anweisungsinhalt besteht aus der Schaltprozedur derjenigen Transaktion, die der Transition eineindeutig zugeordnet ist. Eine Produktionsregel stellt ebenso eine bedingte Anweisung dar: Ihre Bedingung ist die Regelvoraussetzung. Der Anweisungsinhalt wird durch die Reduktions- und Expansionsformeln aus der Regelwirkung ausgedrückt. Die

Anweisungsinhalte einer transitionsspezifischen Schaltregel und einer Produktionsregel entsprechen einander unmittelbar. Denn beide legen fest, daß die Faktenmultimenge der aktuellen Netzmarkierung (Referenzmarkierung) durch das Entfernen alter und das Ergänzen neuer faktischer Formelvorkommnisse in die wohlbestimmte Faktenmultimenge der jeweils hervorgebrachten Folgemarkierung transformiert wird. Zwischen den Anweisungsbedingungen einer transitionsspezifischen Schaltregel und einer Produktionsregel besteht eine ähnliche, allerdings nicht genauso stringente Übereinstimmung[73)]. Daher liegt es nahe, Transitionen von Synthetischen Netzen als Produktionsregeln aufzufassen[74)]. Diesem Ansatz wird durch die später begründete Präferenz zugunsten operationaler Netzmodelle weitgehend gefolgt.

Es wird in dieser Arbeit vorausgesetzt, daß am Ende der Konzeptualisierung eines Modellierungsobjekts stets ein prädikatenlogisches Objektmodell vorliegt, das entweder in Klausel- oder aber in Produktionsregelform notiert wird. Solche deklarativen bzw. operationalen prädikatenlogischen Objektmodelle besitzen eine erhebliche theoretische Bedeutung. Denn sie gestatten es, einerseits die Konstruierbarkeit von Netzmodellen grundsätzlich nachzuweisen und andererseits die Konstruktionsvoraussetzungen offenzulegen. Dennoch handelt es sich um Artefakte, die für die praktische Konstruktion von Netzmodellen im allgemeinen keine wesentlichen Einsichten vermitteln. Daher brauchen sie nicht explizit aufgeführt zu werden. Statt dessen werden sie in dieser Arbeit zumeist implizit vorausgesetzt.

Der Verzicht auf eine explizite Darstellung prädikatenlogischer Objektmodelle läßt sich in zweifacher Weise rechtfertigen. Erstens stellen die deklarativen oder operationalen Objektmodelle nicht das wesentliche Konstruktionsziel des Modellierungsprozesses dar. Es handelt sich nur um Zwischenstufen auf dem Weg zu den intendierten Netzmodellen. Daher besitzen die prädikatenlogischen Objektmodelle in Klausel- oder Produktionsregelform keinen wesentlichen pragmatischen Wert. Zweitens wird jedes prädikatenlogische Objektmodell, das einem Netzmodell in der Gestalt eines Synthetischen Netzes zugrundeliegt, nahezu direkt angezeigt. Denn die Netzspezifikation $SPEC_{MSIG}$ stellt eine algebraisch-prädikatenlogische Spezifikation dar, die nicht nur das Fundament einer algebraischen Markensignatur MSIG offenlegt. Vielmehr läßt sich aus den Komponenten der Netzspezifikation $SPEC_{MSIG}$ auch unmittelbar rekonstruieren, welches prädikatenlogische Objektmodell dem Netzmodell zugrundeliegt[75)]. Infolge dieser Rekonstruktionsmöglichkeit ist es abermals nicht nötig, das prädikatenlogische Objektmodell, auf dem ein Netzmodell aufbaut, explizit anzuführen.

Anmerkungen zum Kapitel:

1) Vgl. dazu die Anmerkungen zu kognitionsbedingten Modellveränderungen. Vgl. ebenso das Beispiel für eine kognitionsbedingte Modellerweiterung, die für ein deklaratives Objektmodell einen neuen Zustand hervorbringt.

2) Aus diesem Leistungsvermögen wurde die Bezeichnung "deklaratives Objektmodell" abgeleitet. Denn die deklarative Semantik einer Formelmenge wurde als Menge aller logischen Konsequenzen definiert, die aus dieser Formelmenge gefolgert werden können.

3) Prädikatenlogische Formeln, welche die hier definierte konjunktive Normalform erfüllen, werden fortan vereinfacht als konjunktive Normalforme(l)n bezeichnet.

4) Die vollständige Notation des Formelvorkommnisses $prä_{u(p.q)}$ lautet $prä_{u(p.q)}(te_{1.p.q},...,te_{Ku.p.q})$. Der Übersichtlichkeit halber wird auf die präzisierende Indexierung zugunsten der o.a. einfacheren Notation verzichtet, solange keine Mißverständnisse zu befürchten sind. Wegen $K_u \in \mathcal{N}_0$ sind auch 0-stellige Formelvorkommnisse $prä_{u(p.q)}()$ zulässig, die jeweils den Charakter einer zustandsbeschreibenden Aussage besitzen. Es wird jedoch vorausgesetzt, daß keines der K_u-stelligen Formelvorkommnisse die tautologische Formel "T" oder die kontradiktorische Formel "$\perp$" darstellt.

5) Da diese Allquantoren nicht explizit notiert werden, können Literale vereinfacht als *atomare* Formelvorkommnisse oder deren Negate angesprochen werden. Strenggenommen stellt jedoch auch ein positives Literal ein zusammengesetztes Formelvorkommnis dar, sobald das Formelargument mindestens eine Variable enthält und der Formel ein Allquantor explizit vorangestellt ist (vgl. dazu die Definition zusammengesetzter Formeln). Vgl. zur vorherrschenden impliziten Notation von Allquantoren bei Verwendung der konjunktiven Normalform MURATA,TA. (1988b), S. 482; CORDES (1988), S. 32 u. 34.

6) Wenn eine Klausel sowohl ein atomares Formelvorkommnis $prä_{u(p.q)}$ als auch dessen Negat $\neg prä_{u(p.q)}$ umfaßt, enthält sie aufgrund ihrer internen adjunktiven Literaleverknüpfung das tautologische Adjugat $prä_{u(p.q)} \vee (\neg prä_{u(p.q)})$. Ein solcher tautologischer Klauselbestandteil entspricht aus materieller Perspektive einer Nebenbedingung. Sie führt in Netzmodellen entweder einer 1-Schleife oder zu einer Informationskante.

7) Dies läßt sich anhand einer Fallunterscheidung verdeutlichen. Der erste Fall bezieht sich sowohl auf positive Literale, die aus tautologischen Formelvorkommnissen bestehen, als auch auf negative Literale, die durch kontradiktorische Formelvorkommnisse konstituiert werden. Jedes dieser Literale ist allgemeingültig. Eine atomare Klausel, die nur aus einem solchen allgemeingültigen Literal besteht, ist ebenso allgemeingültig. Gleiches gilt für jede zusammengesetzte Klausel, die ein allgemeingültiges Literal enthält. Denn ein Adjugat, das mindestens eine tautologische Formel umfaßt, ist unter jeder beliebigen Interpretation des Objektmodells gültig. Der Wahrheitswert eines Konjugats wird durch eine allgemeingültige Konjugatskomponente nicht beeinflußt. Folglich ist eine Klausel, die aus mindestens einem allgemeingültigen Literal aufgebaut wurde, für den Wahrheitswert einer Formel $prä_{FS}$ in konjunktiver Normalform unbeachtlich. Daher brauchen tautologische Formelvorkommnisse in positiven Literalen und kontradiktorische Formelvorkommnisse in negativen Literalen bei der Objektmodellierung nicht berücksichtigt zu werden.
Der zweite Fall erstreckt sich einerseits auf positive Literale, die aus kontradiktorische Formelvorkommnissen gebildet werden, und andererseits auf negative Literale, die aus tautologischen Formelvorkommnissen bestehen. Jedes dieser Literale stellt eine inkonsistente Formel dar. Eine atomare Klausel, die nur aus einem solchen inkonsistenten Literal besteht, ist ebenso inkonsistent. Eine inkonsistente Klausel zieht die schwerwiegende Konsequenz nach sich, daß die gesamte objektmodellierende Formel $prä_{FS}$ in konjunktiver Normalform inkonsistent ist. Denn ein Konjugat kann niemals gültig sein, sobald es mindestens eine immer ungültige Teilformel enthält. Es wurde jedoch an früherer Stelle vorausgesetzt, daß für die Abbildung von Realproblemen nur konsistente Objektmodelle benutzt werden. Vgl. dazu die Anmerkungen zur Konsistenzanforderung. Daher lassen sich inkonsistente atomare Klauseln von vornherein ausscheiden. Ein inkonsistentes Literal kann aber auch Bestandteil einer zusammengesetzten Klausel sein. Der Wahrheitswert eines Adjugats hängt aber in keiner Weise von einer inkonsistenten Adjugatskomponente ab. Daher ist auch jedes inkonsistente Literal, das an der adjunktiven Zusammensetzung einer Klausel teilnimmt, für den Wahrheitswert einer Formel $prä_{FS}$ in konjunktiver Normalform unbeachtlich. Insgesamt brauchen also weder kontradiktorische Formelvorkommnisse in positiven Literalen noch tautologische Formelvorkommnisse in negativen Literalen bei der Objektmodellierung beachtet zu werden.

8) Wenn keine abweichenden Festlegungen erfolgen, wird unter einer Klausel stets eine typische Klausel verstanden.

9) Gemeint ist der starke prädikatenlogische Äquivalenzbegriff: Die Bijugate "$(prä :\Leftrightarrow \perp) \leftrightarrow (KL = \{\})$" und "$(prä :\Leftrightarrow kla_1 \wedge ... \wedge kla_P) \leftrightarrow (KL = \{kla_1,...,kla_P\})$" sind jeweils allgemeingültig.

10) Vgl. ESSER,H. (1977a), S. 37a, Fall 8 u. 14; BUCHER (1987), S. 105, Regel 11 u. 12.

11) Vgl. ESSER,H. (1977a), S. 37a, Fall 11; WECK (1982), S. 72; BUCHER (1987), S. 108, Regel 20.

12) Vgl. BUCHER (1987), S. 106, Regel 15.

13) Darin liegt keine Einschränkung der universellen Anwendbarkeit von Klauseln. Denn jedes Subjugat, das in seinem (konjunktiv zusammengesetzten) Antezedens oder in seiner (adjunktiv zusammengesetzten) Konklusion mindestens eine negierte atomare Formel enthält, läßt sich in ein äquivalentes, aber negatfreies Subjugat transformieren, das nur noch aus atomaren Formeln aufgebaut ist. Diese Transformation läßt sich prinzipiell immer dadurch realisieren, daß das erstgenannte Subjugat zunächst mittels der definitorischen Rückführung aller Subjugate auf Adjugate als eine Klausel dargestellt wird. Diese Klausel wird anschließend in der oben beschriebenen Weise in ein negatfreies Subjugat transformiert. Diese Vorgehensweise wird anhand zweier einfacher Beispiele demonstriert, die jeweils eine negierte atomare Formel im Antezedens oder in der Konklusion eines Subjugats enthalten:

$$\begin{aligned}
& (\text{prä}_1 \wedge (\neg\text{prä}_2) \wedge \text{prä}_3) \rightarrow (\text{prä}_4 \vee \text{prä}_5) \\
\Leftrightarrow\ & \neg(\text{prä}_1 \wedge (\neg\text{prä}_2) \wedge \text{prä}_3) \vee (\text{prä}_4 \vee \text{prä}_5) \\
\Leftrightarrow\ & (\neg\text{prä}_1) \vee (\neg(\neg\text{prä}_2)) \vee (\neg\text{prä}_3) \vee \text{prä}_4 \vee \text{prä}_5 \\
\Leftrightarrow\ & (\neg\text{prä}_1) \vee (\neg\text{prä}_3) \vee \text{prä}_2 \vee \text{prä}_4 \vee \text{prä}_5 \\
\Leftrightarrow\ & \neg(\text{prä}_1 \wedge \text{prä}_3) \vee (\text{prä}_2 \vee \text{prä}_4 \vee \text{prä}_5) \\
\Leftrightarrow\ & (\text{prä}_1 \wedge \text{prä}_3) \rightarrow (\text{prä}_2 \vee \text{prä}_4 \vee \text{prä}_5)
\end{aligned}$$

$$\begin{aligned}
& (\text{prä}_1 \wedge \text{prä}_2) \rightarrow (\text{prä}_3 \vee (\neg\text{prä}_4) \vee \text{prä}_5) \\
\Leftrightarrow\ & \neg(\text{prä}_1 \wedge \text{prä}_2) \vee (\text{prä}_3 \vee (\neg\text{prä}_4) \vee \text{prä}_5) \\
\Leftrightarrow\ & (\neg\text{prä}_1) \vee (\neg\text{prä}_2) \vee \text{prä}_3 \vee (\neg\text{prä}_4) \vee \text{prä}_5 \\
\Leftrightarrow\ & (\neg\text{prä}_1) \vee (\neg\text{prä}_2) \vee (\neg\text{prä}_4) \vee \text{prä}_3 \vee \text{prä}_5 \\
\Leftrightarrow\ & \neg(\text{prä}_1 \wedge \text{prä}_2 \wedge \text{prä}_4) \vee (\text{prä}_3 \vee \text{prä}_5) \\
\Leftrightarrow\ & (\text{prä}_1 \wedge \text{prä}_2 \wedge \text{prä}_4) \rightarrow (\text{prä}_3 \vee \text{prä}_5)
\end{aligned}$$

14) Falls es sich bei den Antezedensformeln um variable Formeln handelt, ist jede ihrer Variablen - zumindest implizit - durch einen Allquantor gebunden. Vgl. zu dieser Allquantifizierung von Antezedensformeln FRONHÖFER (1987), S. 16 (im Kontext der logischen Programmierung von Linearen Plänen). Vgl. auch die entsprechende Allquantifizierung der Variablen aus HORN-Formeln.

15) Sofern eine Konklusionsformel als eine variable Formel vorliegt, ist jede ihrer Variablen - zumindest implizit - wiederum durch einen Allquantor gebunden. Dies entspricht abermals der Allquantifizierung der Variablen aus HORN-Formeln. Dagegen wird der logischen Programmierung von Linearen Plänen nicht gefolgt. Denn dort erfolgt eine Existenzquantifizierung von Konklusionsformeln; vgl. FRONHÖFER (1987), S. 16.

16) Eine Vereinfachung liegt insofern vor, als die Konklusion einer subjunktiven Klauseln kla_p mit $Q_{P,p}=1$ kein Adjugat, sondern ein einzelnes atomares Formelvorkommnis darstellt.

17) Nachfolgend wird nur ein schwacher Äquivalenzbegriff verwendet: Das Formelsystem FS einerseits und seine Klauselmenge KL_{FS} oder konjunktive Normalform $prä_{FS}$ andererseits gelten genau dann als "äquivalent", wenn die Lösungen von prädikatenlogischen Entscheidungsprobleme nicht davon abhängen, ob die Probleme als Formelsysteme FS bzw. als konjunktive Normalformen $prä_{FS}$ oder Klauselmengen KL_{FS} ausgedrückt werden. Dies bedeutet inhaltlich, daß sich die Allgemeingültigkeit und Inkonsistenz von Formeln invariant verhalten, wenn zwischen den ursprünglichen Formeln des Formelsystem FS auf der einen Seite und den transformierten Formeln der konjunktiven Normalform $prä_{FS}$ oder der Klauselmenge KL_{FS} auf der anderen Seite gewechselt wird. Der starke prädikatenlogische Äquivalenzbegriff würde dagegen erfordern, daß nicht nur die beiden o.a. semantischen Formelqualitäten erhalten bleiben, sondern darüber hinaus auch alle Formelwahrheitswerte. Dies kann jedoch von der nachfolgend skizzierten Transformation von Formelsystemen in deren konjunktiven Normalformen und Klauselmengen im allgemeinen nicht garantiert werden. Ursache dafür sind die Skolemisierungsoperationen.

18) Vgl. BÖHRINGER (1988), S. 15.

19) Die nachfolgenden Transformationen lehnen sich an die Erzeugung konjunktiver Normalformen an, wie sie z.B. von CORDES (1988), S. 32ff.; BÖHRINGER (1988), S. 14f.; CLOCKSIN (1990), S. 268ff., beschreiben wird. Allerdings stellt die Transformation 0-stelliger in 1-stellige Formeln eine Besonderheit dar, die in den vorgenannten Quellen nicht behandelt wird. Denn in prädikatenlogischen Kontexten wird die Möglichkeit, einfache Sachverhalte durch 0-stellige Prädikatssymbole zu repräsentieren, vernachlässigt. Bei CLOCKSIN (1990), S. 302ff., findet sich eine ausführlich beschriebene, unmittelbar auf Automatischen Informationsverarbeitungssystemen ausführbare Technik, um prädikatenlogische Formeln in konjunktiver Normalform darzustellen. Sie beruht auf dem gleichen Transformationskonzept wie die vorgenannten Quellen und die oben erläuterte Formeltransformation.

20) Diese Formeln dürfen sowohl atomar als auch beliebig komplex zusammengesetzt sein.

21) Hiermit ist der strenge prädikatenlogische Äquivalenzbegriff gemeint: Das Bijugat "$FS = \{prä_h: h=1,...,H_k \wedge ... H_k \geq 2\} \leftrightarrow prä_{FS}^* :\Leftrightarrow prä_1 \wedge ... \wedge prä_{Hk}$" ist allgemeingültig.

22) Die Konjugatformel wird als $prä_{FS}^*$ notiert, da sie noch keineswegs die intendierte konjunktive Normalform $prä_{FS}$ erfüllen muß.

23) Diese Formel kann wiederum atomar oder auch beliebig komplex zusammengesetzt sein.

24) Die Skolemisierung von Formeln mit Existenzquantoren wurde bereits angesprochen. Vgl. zu den technischen Details der Operationsausführung die dort angeführten Quellen, z.B. die besonders anschauliche Erläuterung von CORDES (1988), S. 33f., und CLOCKSIN (1990), S. 269. Eine detaillierte Formulierung der Skolemisierungsoperation, die in der Programmiersprache PROLOG abgefaßt ist und sich unmittelbar implementieren läßt, bietet CLOCKSIN (1990), S. 304 i.V.m. S. 183ff., an.
Es wurde auch schon darauf hingewiesen, daß die Formelskolemisierungen keine Äquivalenztransformationen darstellen. Es reicht jedoch für prädikatenlogische Analysen von Formelsystemen aus, daß die Allgemeingültigkeit und Inkonsistenz von Formeln gegenüber diesen Skolemisierungsoperationen invariant sind. Auch dies wurde schon dargelegt.

25) Allerdings zeigt die voranstehende Transformation von Negaten, daß sich Existenzquantoren auch dann in ein prädikatenlogisches Objektmodell "einschleichen" können, wenn sie ursprünglich für die Beschreibung des Modellzustands überhaupt nicht verwendet wurden. Denn das Negat einer allquantifizierten Formel wird in eine Formel transformiert, in der einem vereinfachten Negat ein Existenzquantor vorangestellt wird. Um auch diesen Sonderfall beherrschen zu können, kommen zwei Auswege in Betracht: Entweder werden Skolemisierungsoperationen angewendet, um solche nachträglich aufgetretenen Existenzquantoren wieder zu eliminieren. Oder die Quantifizierungs-Prämisse wird so erweitert, daß der o.a. Sonderfall überhaupt nicht eintreten kann. Dann muß zusätzlich gefordert werden, daß in der prädikatenlogischen Zustandsbeschreibung durch das Formelsystem FS keine Formel enthalten sein darf, die erstens durch einen Allquantor eingeleitet und zweitens in ihrer Gesamtheit negiert wird. Der Verf. bevorzugt den zweiten Ansatz. Daher vermeidet er bei der prädikatenlogischen Zustandsbeschreibung grundsätzlich negierte Allformeln (erweiterte Quantifizierungs-Prämisse). Vgl. zur Problematik von Existenzquantoren auch die eng verwandten Ausführungen zu potentiellen Fehlschlüssen, die im Rahmen der operationalen Dimension der Prädikatenlogik angesprochen wurden.

26) Diese Transformation ist notwendig, um prädikatenlogische Formelsysteme in Netzmodelle übersetzen zu könne, die mit Höheren Netzen arbeiten. Denn in solchen Netzmodellen können die Gültigkeit von Formeln und die Veränderung von Formelgültigkeiten nur mit der Hilfe von Marken dargestellt werden, welche die Formelargumente konstituieren. 0-stellige Formeln besitzen aber per definitionem keine Argumente. Dennoch können sie gültig sein - dann stellen sie wahre Aussagen dar - und auch ihren Wahrheitswert verändern. Daher muß für eine vollständige Erfassung aller prädikatenlogischen Formeln eine Konstruktion angeboten werden, welche die Gültigkeit von 0-stelligen Formeln mit der Hilfe von Marken erklärt. Genau dies geschieht oben durch die 1-stelligen Substitutformeln, die aus dem Prädikatssymbol $Prä_u$(bas_marke) abgeleitet sind. Ihre Gültigkeit wird mit Hilfe der Kopie "$\varnothing$" der Basismarke in impliziter Weise als $prä_u(\varnothing)$ oder in expliziter Weise als $fakt_f(prä_u(\varnothing))$ repräsentiert.

27) Es könnte auch die Substitutformel prä(X) verwendet werden. Der Definitionsbereich der Variablen X wäre dann aber die einelementige Menge $OB_{bas\_marke} = \{\varnothing\}$, die nur die Kopie der Basismarke umfaßt. Da für die Belegung der Variablen "X" per constructionem keine Wahlmöglichkeiten bestehen, unterscheidet sie sich von der oben benutzten, konstanten Substitutformel prä($\varnothing$) nur in formaler, aber nicht in materieller Hinsicht.

28) Ohne den Zwischenschritt der konjunktiven Normalform verfährt z.B. GENRICH (1990b) bei seinen exemplarischen Zuordnungen zwischen prädikatenlogischen Formeln und zugehörigen Klauselmengen. Allerdings legt er die hierbei implizit durchgeführte Formeltransformation nicht offen.

29) Zwei dieser Normalformen wurden bereits mehrfach angesprochen: die konjunktive und die skolemisierte Normalform. Komplementär zur konjunktiven läßt sich auch eine adjunktive Normalform bilden.

30) Daran knüpft die Bezeichnung "operationales Objektmodell" an: Sie verweist auf die "operationsbedingten" Veränderungen von modellierten Objekten. Sie treten immer dann ein, wenn Ausführungen realer Aktionen in einem Objektmodell durch entsprechende Operationsausführungen abgebildet werden.

31) Vgl. zur zustandsbezogenen Interpretation von Netzmarkierungen HAAS (1987), S. 24; LU,M. (1987), S. 108f.; FREEDMAN (1988b), S. 334.

32) Diese faktenbezogene Interpretation von Netzmarkierungen wird mitunter auch vereinfacht dadurch ausgedrückt, daß die Markierungen eines Netzmodells Formeln darstellen. Dabei wird vom metasprachlichen Aspekt der - präsupponierten - Formelgültigkeit ebenso abstrahiert wie von der Fokussierung auf atomare Formelvorkommnisse.

Vgl. zu dieser fakten- oder formelbezogenen Markierungsinterpretation MARTI-OLIET (1989), S. 321 ("identifying propositions with markings").
Wenn die faktenbezogene Interpretation von Netzmarkierungen zugrundegelegt wird, gilt für das Wissen, das von einem operationalen Objektmodell ausgedrückt wird: Das explizite Wissen erstreckt sich einerseits auf die Menge aller Fakten, die als gültige atomare Formelvorkommnisse den Ausgangszustand des Objektmodells beschreiben (Zustandswissen). Es umfaßt andererseits die Menge aller zulässigen Veränderungen von Faktenmengen, die von den Produktionsregeln spezifiziert werden (Aktionswissen). Das implizite Wissen des operationalen Objektmodells erstreckt sich auf alle Faktenmengen, die sich aus der Faktenmenge für den Ausgangszustand des Objektmodells ableiten lassen. Dabei wird die Ableitbarkeit von Faktenmengen durch Produktionsregeln des operationalen Objektmodells festgelegt. Die Produktionsregeln werden ihrerseits im zugehörigen Netzmodell als Transitionen wiedergegeben. Daher gilt für die Markierungen, die sich im Netzmodell von dessen Ausgangsmarkierung aus durch Schaltakte von Transitionen erreichen lassen: Sie geben alle Faktenmengen wieder, die im zugrundeliegenden operationalen Objektmodell durch das Explizieren seines impliziten Wissens abgeleitet werden können. Daher korrespondiert die Erreichbarkeit von Markierungen im Netzmodell mit der Ableitbarkeit von Faktenmengen für das zugrundeliegende operationale Objektmodell. Vgl. zu dieser Korrespondenz MARTI-OLIET (1989), S. 321 (dort werden allerdings die Faktenmengen nur als Formeln angesprochen).
Gegenüber der Explizierung von implizitem Wissen, die auch bei deklarativen Objektmodellen erfolgen kann, bestehen zwei fundamentale Unterschiede. Sie hängen beide damit zusammen, daß in einem deklarativen Objektmodell immer nur das - explizite oder implizite - Wissen für genau einen Modellzustand repräsentiert wird. Daher wird in einem deklarativen Objektmodell einerseits lediglich zustandsbeschreibendes Wissen ausgedrückt. Wissen über Zustandstransformationen, das in einem operationalen Objektmodell als Aktionswissen vorgehalten wird, ist daher in seinem deklarativen Pendant grundsätzlich nicht vorgesehen. Andererseits erstreckt sich das implizite Zustandswissen eines deklarativen Objektmodells auf alle logischen Konsequenzen, die aus dem einen repräsentierten, stets *unveränderten* Modellzustand gefolgert werden können. Das implizite Wissen eines operationalen Objektmodells wird dagegen abgeleitet, indem der Ausgangszustand des Objektmodells durch das Ausführen von Produktionsregeln in zulässige Folgezustände transformiert wird. Dieses implizite Wissen betrifft daher *veränderte* Modellzustände. Darüber hinaus stellt es kein reines Zustandswissen dar. Vielmehr handelt es sich bei dem impliziten Wissen eines operationalen Objektmodells wegen der zustandstransformierenden Regelausführungen um ein kombiniertes Zustands- und Aktionswissen.

33) Strenggenommen gilt diese konjunktive Elementverknüpfung nur, falls die Mengen jeweils mindestens zwei Elemente enthalten. Dies ist aber der Regelfall.

34) In einer früheren Erläuterung wurden die Komponenten der Interpretation I_0 für das Formelsystem FS eins prädikatenlogischen Objektmodells aufgelistet.

35) In der Literatur, die sich mit Produktionsregeln befaßt, ist dies zumeist der Fall. Ein Grund dafür mag sein, daß die Konzentration auf Produktionsregeln eine Minderbeachtung der prädikatenlogischen Semantik nach sich zieht.

36) Dies gilt nur, sofern keine deklarativen, sondern operationale Objektmodelle angestrebt werden. An früherer Stelle wurde jedoch schon dargelegt, daß der Verf. operationale Objektmodelle bevorzugt.

37) Das Konzept der Produktionsregeln wurde erstmals von POST,E. (1943), S. 197ff., insbesondere S. 199ff., formuliert und in einen formalen Kalkül - ein Produktions(regel)system - eingebettet. Es findet derzeit als Formalismus für die Wissensrepräsentation von Expertensystemen breite Beachtung. In diesem Zusammenhang hat es der Verf. an anderer Stelle schon ausführlicher dargestellt; vgl. ZELEWSKI (1986a), S. 199ff. Vgl. darüber hinaus auch DAVIS,R. (1975), S. 1ff.; BÖHRINGER (1988), S. 203ff.
Einen kompakten Überblick über Produktionsregeln, der ihren nachfolgend erläuterten subjunktiven Charakter besonders klar erkennen läßt, bietet z.B. aus dem Kreis der vorgenannten Quellen BÖHRINGER (1988). Auf Produktionsregelsysteme wird später noch ausführlicher eingegangen; vgl. dazu die Ausführungen zur Spezifizierung der Schaltvorschriften von Transitionen durch Produktionsregelgruppen.

38) Auf eine Ausnahme wird weiter unter hingewiesen; vgl. dazu die Erläuterung unbedingter Produktionsregeln.

39) Daher wird auch von bedingten Produktionsregeln gesprochen.

40) Diese Ambivalenz trägt zum großen Ausdrucksreichtum des Produktionsregelkonzepts bei, aufgrund dessen es vom Verf. gegenüber den rein deklarativen prädikatenlogischen Formelsystemen vorgezogen wird.

41) Eine ähnliche Produktionsregelnotation findet sich immerhin bei HOLZMANN (1987), S. 340, als "transition rule". Allerdings wird dort die Regel nur auf der obersten Ebene von Regelvoraussetzung und -wirkung behandelt. Weiterführende Verfeinerungen der beiden Regelkomponenten finden nicht statt. Darüber hinaus werden Regelvoraussetzung und -wirkung nicht in subjunktiver Weise verknüpft, sondern einander nebengeordnet.

42) In der vorherrschenden Notation von Produktionsregeln werden dagegen ihr deklarativer und operationaler Charakter miteinander vermengt. Dies wird besonders deutlich, wenn Produktionsregelsysteme als Termersetzungs-

systeme ausgestaltet werden. Vgl. zu solchen Termersetzungssystemen z.B. HSIANG (1983a), S. 333ff.; DROSTEN (1989), S. 4ff. u. 22ff.
In Termersetzungssystemen und allen ähnlich notierten Produktionsregelsystemen wird nur zwischen einer "linken" und einer "rechten" Regelhälfte unterschieden. Bei jeder Regelanwendung wird der Ausdruck ("Term"), der die linke Regelhälfte bildet, durch den Ausdruck der rechten Regelhälfte ersetzt. Der Übergang vom links zum rechts notierten Ausdruck stellt den operationalen Aspekt der Regelwirkung dar. Der deklarative Aspekt der Regelvoraussetzung wird durch die linke Regelhälfte konstituiert: Ihr Ausdruck muß vorliegen, damit die Ausdrucksersetzung überhaupt ausgeführt werden kann. Die gleiche Interpretation von "linker" und "rechter" Regelhälfte liegt dem Konzept Linearer Beweise zugrunde, auf das später zurückgekommen wird. Vgl. zu den Festlegungen für die dort verwendeten aktionsbeschreibenden Regeln FRONHÖFER (1987), S. 16 (mit invertierten Subjugaten); BIBEL (1989), S. 52.
Die Vermengung von operationalem und deklarativem Charakter äußert sich vor allem darin, daß die linke Regelhälfte nicht nur zur operationalen Definition der Regelwirkung gehört. Vielmehr konstituiert sie auch die deklarative Regelvoraussetzung. Diese Konfundierung von deklarativer und operationaler Regelkomponente in der linken Regelhälfte zieht eine bemerkenswerte Verengung des Ausdruckspotentials von derart notierten Produktionsregeln nach sich. Denn es ist auf diese Weise unmöglich, in deklarativer Weise eine Regelvoraussetzung zu formulieren, die von der Operationsausführung der Regelwirkung nicht betroffen wird: Der Ausdruck aus der linken Regelhälfte, der die Regelvoraussetzung spezifiziert, wird per constructionem bei jeder Regelanwendung eliminiert, indem er durch den Ausdruck aus der rechten Regelhälfte ersetzt wird. Falls die Regelvoraussetzung, die für eine Regelanwendung erfüllt sein muß, durch die Regelwirkung nicht aufgehoben werden soll, muß die Eliminierung des Ausdrucks aus der linken Regelhälfte in der Regelwirkung wieder zurückgesetzt werden. Diese Fehlerkompensation entspricht genau der Modellierung von Nebenbedingungen in Stelle/Transition-Netzen durch 1-Schleifen. Die dort vorgetragene Kritik trifft ebenso auf die oben vorgestellte übliche Notation von Produktionsregeln zu. Die vorgenannten Schwierigkeiten vermeidet der Verf. von vornherein. Zu diesem Zweck wird in der nachfolgend vorgestellten Produktionsregelnotation die deklarative Regelvoraussetzung von der operationalen Regelwirkung grundsätzlich getrennt. Auf das oben skizzierte Konfundierungsproblem der konventionellen Produktionsregelnotation wird an anderer Stelle vertiefend zurückgekommen.

43) Genau das ist bei der sonst üblichen Notation von Produktionsregeln nicht der Fall. Denn dort stellt das Antezedens deines Produktionsregel die "linke" Regelhälfte dar, in der deklarative Regelvoraussetzung und operationale Regelwirkung miteinander vermengt werden. Vgl. dazu die voranstehende Anmerkung. Nur bei HOLZMANN (1987), S. 340, beschränkt sich die Regelvoraussetzung auf eine rein deklarativ definierte, prädikatenlogische Formel.

44) Auch dies trifft bei konventionell notierten Produktionsregeln nicht zu. Dort wird die Regelwirkung spezifiziert, indem "linke" und "rechte" Regelhälfte miteinander verknüpft werden. Wiederum stimmt HOLZMANN (1987), S. 340, mit der hier bevorzugten Regelnotation überein. Auch dort wird die Regelwirkung ausschließlich in der rechten Regelhälfte durch eine zustandsverändernde Aktion ausgedrückt.

45) Die Notation "con" verweist darauf, daß die Regelvoraussetzung Bedingungen ("conditions") formuliert, welche für die Ausführung der Produktionsregel erfüllt sein müssen.

46) Die Notation "act" erinnert daran, daß die Regelwirkung Aktionen ("actions") aus dem jeweils modellierten Realitätsausschnitt abbildet.

47) Es handelt sich nur um eine notwendige, aber um keine hinreichende Ausführungsvoraussetzung. Dies hat zwei Gründe. Erstens wird bei jeder Produktionsregel implizit vorausgesetzt, daß die Regelausführung niemals zu unzulässigen Faktenmengen führt. Dies gilt auch dann, wenn alle Formelvorkommnisse aus der Regelvoraussetzung als gültig bekannt sind. Solche ungültigen Faktenmengen würden immer dann vorliegen, wenn die Anzahlen zugehöriger Fakten entweder negativ würden oder die höchstzulässigen Faktenanzahlen überstiegen. Beide Fälle werden durch die Formulierung von Produktionsregeln jedoch nicht explizit erfaßt. Hierin liegt ein Mangel des Produktionsregelkonzepts. Dieses Explizierungsdefizit wird hier nicht weiter verfolgt, weil Produktionsregeln in operationalen Netzmodellen durch Transitionen abgebildet werden, deren Aktivierungsbedingungen sicherstellen, daß niemals unzulässige Faktenmengen hervorgebracht werden können. In dieser Hinsicht leistet das Petrinetz-Konzept einen bemerkenswerten Beitrag zur Offenlegung einer wesentlichen, aber nur mangelhaft explizierten Eigenschaft des Produktionsregelkonzepts. Zweitens brauchen Produktionsregeln keineswegs ausgeführt zu werden, falls ihre Regelvoraussetzungen erfüllt sind - und sichergestellt ist, daß keine unzulässigen Faktenmengen produziert werden. Denn dann ist die Regelausführung nur erlaubt, aber keineswegs obligatorisch. Dies entspricht genau der Permissivität der Schaltregeln für Petrinetze. In beiden voranstehend erörterten Hinsichten erweist sich die überaus enge konzeptionelle Verwandtschaft zwischen Produktionsregelsystemen und Petrinetzen.

48) Nicht-tautologische Voraussetzungsformeln werden fortan auch kurz als Voraussetzungsformeln bezeichnet, wenn aus dem Argumentationskontext oder der Notation $V_y \in \mathcal{N}_+$ ersichtlich ist, daß die tautologische Formel ausgeschlossen wird.

49) Die vollständige Notation der Formelvorkommnisse lautet $prä_{u(y.v)}(te_{1.y.i.d},...,te_{Ku.y.i.d})$. Der Übersichtlichkeit halber wird aber die Indizierung der Argumentterme in der o.a. Weise vereinfacht, solange keine Mißverständnisse zu befürchten sind. Dies gilt für alle nachfolgenden Ausführungen zu Produktionsregeln.

50) Es wird vorausgesetzt, daß jede Produktionsregel pr_y für jedes Prädikatssymbol $Prä_u$ in der Voraussetzungsmenge CON_y höchstens eine Inklusionsformel $ink_{y.v}$ enthält, die sich in der oben definierten Weise auf das Prädikatssymbol $Prä_u$ bezieht. Dies Prämisse wirkt nicht restriktiv, weil die Verwendung der Multimengen $MTAI_{u(y.v).n(y)}$ gestattet, beliebig viele Formelvorkommnisse $prä_{u(y.v)}(te_1,...,te_{Ku})$ des Prädikatssymbols $Prä_u$ in die Inklusionsformel $ink_{y.v}$ einzubeziehen.

51) Die Grundterme stellen in der Regel Konstanten dar. Es ist aber nicht ausgeschlossen, daß eine Variable auch durch einen komplexen Grundterm aus mindestens einem Operator und mindestens einer Konstanten gebunden wird.

52) Die Variablenbindungsfunktionen vb_c bedeuten eine technische Erleichterung für die Formulierung von Produktionsregeln. Wenn eine Regelwirkung für unterschiedliche, aber strukturell gleichartige Regelvoraussetzungen gilt, brauchen nicht entsprechend viele verschiedene Produktionsregeln definiert werden. Statt dessen ist es oftmals möglich, die Regelvoraussetzung mit der Hilfe von Variablen in den Argumenten der Formelvorkommnisse aus den o.a. Multimengen so flexibel zu formulieren, daß eine Produktionsregel ausreicht, um alle vorgenannten gleichartigen Voraussetzungsvarianten abzudecken. Jede Variablenbindungsfunktion, die alle Variablen in der Regelvoraussetzung auf Grundterme abbildet, entspricht dann genau einer Voraussetzungsvariante.

53) Dadurch wird sichergestellt, daß die Variablen der algebraischen Bedingungsformeln durch die Variablenbindungsfunktionen vb_c, die auf Multimengen angewendet werden, eindeutig determiniert sind. Zugleich folgt daraus, daß algebraische Bedingungsformeln nur dann definiert sein können, wenn die Regelvoraussetzung mindestens eine faktenbezogene Inklusionsformel enthält oder wenn die Regelwirkung mindestens eine Reduktionsformel umfaßt.

54) Es handelt sich - wie auch sonst in dieser Arbeit - um ein "oder" im inklusiven Sinn der logische Adjunktion.

55) Dies entspricht der Allquantifizierung aller Variablen, die in den Antezedensformeln von subjunktiven Klauseln vorkommen.

56) Es wird vorausgesetzt, daß jede Produktionsregel pr_y für jedes Prädikatssymbol $Prä_u$ in der Wirkungsmenge ACT_y höchstens eine Reduktionsformel $red_{y.w}$ enthält, die sich in der oben definierten Weise auf das Prädikatssymbol $Prä_u$ bezieht. Diese Prämisse wirkt nicht restriktiv, weil die Verwendung der Multimengen $MTAV_{u(y.w).n(y)}$ gestattet, beliebig viele Formelvorkommnisse $prä_{u(y.w)}(te_1,...,te_{Ku})$ des Prädikatssymbols $Prä_u$ in die Reduktionsformel $red_{y.w}$ einzubeziehen.

57) Dabei darf es sich um kein Prädikatssymbol handeln, das bereits in der Regelvoraussetzung benutzt wurde. Es muß also gelten: $u(y.w) \neq u(y.v)$ für alle $w \in \{1,...,W_y\}$ und alle $v \in \{1,...,V_y\}$, sofern $V_y \in \mathcal{N}_+$. (Für $V_y=0$ enthält die Regelvoraussetzung nur die tautologische Formel, während alle Wirkungsformeln nicht-tautologische Formeln darstellen. Daher entfällt für diesen Fall die vorgenannte Bedingung.) Die Anforderung $u(y.w) \neq u(y.v)$ bedeutet, daß die beiden Prädikatssymbolemengen für die Regelvoraussetzung auf der einen und für die Regelwirkung auf der anderen Seite disjunkt ausfallen. Diese Disjunktheit ist für Produktionsregeln nicht unbedingt erforderlich, trägt aber zu deren klarer Strukturierung bei. Darüber hinaus bereitet die Disjunktheitsforderung die Transformation von operationalen Objektmodellen in entsprechende Netzmodelle vor.

58) Es muß sich um dieselbe Variablenbindungsfunktion vb_c handeln, die schon oben in der Regelvoraussetzung angewendet wird. Dies gilt auch für alle nachfolgenden Vorkommnisse der Variablenbindungsfunktion vb_c.

59) Es wird vorausgesetzt, daß jede Produktionsregel pr_y für jedes Prädikatssymbol $Prä_u$ in der Wirkungsmenge ACT_y höchstens eine Expansionsformel $exp_{y.w}$ enthält, die sich in der oben definierten Weise auf das Prädikatssymbol $Prä_u$ bezieht. Dies Prämisse wirkt nicht restriktiv, weil die Verwendung der Multimengen $MTAN_{u(y.w).n(y)}$ gestattet, beliebig viele Formelvorkommnisse $prä_{u(y.w)}(te_1,...,te_{Ku})$ des Prädikatssymbols $Prä_u$ in die Expansionsformel $exp_{y.w}$ einzubeziehen.

60) Es gilt - wie schon bei den Reduktionsformeln - die Anforderung, daß die Prädikatssymbolemengen für die Regelvoraussetzung und die Regelwirkung disjunkt sind: $u(y.w) \neq u(y.v)$ für alle $w \in \{1,...,W_y\}$ und alle $v \in \{1,...,V_y\}$. Allerdings kann für dasselbe Prädikatssymbol in der Regelwirkung sowohl eine Reduktions- als auch eine Expansionsformel definiert sein. Dies führt später in dem zugehörigen Netzmodell zu einer 1-Schleife. Diese 1-Schleife braucht jedoch keine Nebenbedingung darzustellen. Beispielsweise kann sie auch dazu führen, daß die Attributausprägungen der Kopie einer Attributmarke verändert werden.

61) Daraus folgt, daß algebraische Bestimmungsformeln nur dann definiert sein können, wenn die Regelwirkung mindestens eine faktenbezogene Expansionsformel enthält, deren Argument mindestens eine Variable umfaßt.

62) Hierdurch wird sichergestellt, daß die Wirkung der Regelausführung bei gegebener Variablenbindungsfunktion vb_c eindeutig determiniert ist. Vgl. dazu die detaillierteren, analog gültigen Ausführungen zur Wirkungsdeterminierung des Übergangsschemas ÜS durch entsprechende Bestimmungsformeln.

63) Es handelt sich dabei um eine vereinfachte Sprechweise, da weiterhin eine Regelvoraussetzung - die tautologische Formel - definiert ist.

64) Vgl. zu solchen transitionsbeschriftenden Testbedingungen und Bestimmungsgleichungen z.B. MURATA,TA. (1988b), S. 484.

65) Dies gilt für die algebraischen Bedingungsformeln in der Regelvoraussetzung.

66) Dies trifft auf die algebraischen Bestimmungsformeln in der Regelwirkung zu.

67) Vgl. dazu die Quellen, die zu Produktionsregelsystemen aufgeführt werden; vgl. daraus beispielsweise BÖHRINGER (1988), S. 203.

68) Dies bedeutet keineswegs, daß derart formulierte Produktionsregeln keine Fälle erfassen könnten, in denen vor der Regelausführung die Gültigkeit bestimmter Formelvorkommnisse vorausgesetzt wird, die aufgrund der Regelausführung ihre Gültigkeit verlieren. Die betroffenen Formeln lassen sich ohne Schwierigkeiten als Reduktionsformeln in der o.a. Definition von Produktionsregeln darstellen. Denn für solche Reduktionsformeln müssen vor der Regelausführung gültige Formelvorkommnisse vorliegen, damit die Regelausführung mit ihrer Reduzierung der formelzugehörigen Faktenmenge nicht zu einer unzulässigen Faktenmenge führt. Wenn die Produktionsregel tatsächlich ausgeführt wird, werden die Formelvorkommnisse aus der Faktenmenge eliminiert, verlieren also ihre Gültigkeit. Folglich wird durch die Beschränkung der Regelvoraussetzung auf solche Formelvorkommnisse, die ihre Gültigkeit bei der Regelausführung nicht verlieren, das allgemeine Ausdruckspotential von Produktionsregeln nicht beschnitten.
In den meisten anderen Darstellungen von Produktionsregeln werden dagegen atomaren Formelvorkommnisse, deren Gültigkeit vor der Regelausführung vorausgesetzt wird, die aufgrund der Regelausführung jedoch ihre Gültigkeit verlieren, in abweichender Weise behandelt: Sofern sie überhaupt Berücksichtigung finden, werden sie sowohl in der Regelvoraussetzung als auch in der Regelwirkung aufgeführt. Dabei werden sie in der Regelvoraussetzung als atomare Formelvorkommnisse notiert. In der Regelwirkung erscheinen sie dagegen in ihrer negierten Form. Oder die atomaren Formelvorkommnisse werden dort als ungültig gekennzeichnet. Vgl. dazu beispielsweise die übersichtlichen Ausführungen von BAUMAN (1986), S. 193f.

69) Faktenbezogene Formeln sind die faktenbezogenen Inklusions- und die faktenbezogenen Modifikationsformeln.

70) Bei den algebraischen Formeln handelt es sich um die algebraischen Bedingungs- und um die algebraischen Bestimmungsformeln.

71) Dies ist nicht verwunderlich. Denn Produktionsregeln wurde zuvor bewußt in einer unkonventionellen Weise definiert, um ihre Einbettung in das Konzept Synthetischer Netze zu fördern.

72) Auch die Formulierung der Schaltregel für Stelle/Transition-Netze als eine partielle Funktion besitzt die Qualität einer bedingten Anweisung.

73) Die Übereinstimmung erstreckt sich auf alle Restriktionsformeln aus der Schaltvoraussetzung (Aktivierungsbedingung) einer Transition und auf deren Informationskanten. Sie finden sich in der Regelvoraussetzung einer Produktionsregel als algebraische Bestimmungsformeln bzw. als Inklusionsformeln wieder. Jedoch sind auch zwei Diskrepanzen festzuhalten: Erstens umfassen die algebraischen Bestimmungsformeln auch jene Restriktionsformeln, die zur Schaltwirkung einer Transition gehören. Zweitens umfaßt die Schaltvoraussetzung einer Transition im Aktivierungsprädikat auch die Bedingung, daß alle Markenkopien, die von Eingangsstellen der Transition abgezogen werden sollen, dort vorhanden sein müssen, und daß die Ausgangsstellen der Transition, auf denen Markenkopien abgelegt werden sollen, die dafür erforderliche freie Markenkapazität aufweisen müssen. Diese markenbezogene Komponente der Schaltvoraussetzung einer Transition fehlt in der faktenbezogenen Regelvoraussetzung einer Produktionsregel.

74) Zur Verdeutlichung dieser Auffassung wird eine Transition t_n mit der Aktivierungsbedingung $AKT(t_n,M_r)$ betrachtet. Ihre Schaltwirkung wird durch den Ausdruck "$M_r[t_n\rangle M_f$" notiert Dann läßt sich die transitionsspezifische Schaltregel $SR(t_n)$ als ein Subjugat darstellen, für das gilt:

$$SR(t_n): AKT(t_n,M_r) \rightarrow M_r[t_n\rangle M_f$$

Dies entspricht - vorbehaltlich der Hinweise in der voranstehenden Anmerkung - der Struktur einer Produktionsregel:

$pr_y: con_y \rightarrow act_y$

Die Interpretation der Schaltregel als eine Produktionsregel läßt besonders deutlich werden, daß durch das Schalten der Transition t_n aus einem formalen Objekt - der Referenzmarkierung M_r - ein anderes formales Objekt - die Folgemarkierung M_f - hervorgebracht ("produziert") wird. Hierdurch wird die innere Dynamik eines Schaltakts, das kausale Bewirken einer Markierungsveränderung, formal präzise und zugleich intuitiv anschaulich wiedergegeben. Daher eignen sich Produktionsregeln hervorragend, um den "produktiven" und reflexiven Charakter von Petrinetzen zu unterstreichen, durch das Schalten ihrer Transitionen sich selbst - d.h. die aktuellen Netzmarkierungen - laufend zu verändern.
Die früher vorgestellte Formulierung der Schaltregel von Stelle/Transition-Netzen in der Gestalt einer (partiellen) Funktion verdeckt dagegen diesen produktiven Wirkungszusammenhang auf formaler Ebene. Denn sie konstituiert nur einen funktionalen Zusammenhang zwischen Referenz- und Folgemarkierung. Solche funktionalen Zusammenhänge lassen jedoch keinen zugrundeliegenden Kausalmechanismus erkennen. Denn sie werden in formal derselben Weise ebenso auf akausale Beziehungen angewendet, wie z.B. auf die Zuordnung einer Markenkapazität zu einer Stelle.

75) Die Rekonstruktion ist im allgemeinen eindeutig bestimmt. Denn jedes markenbehaftete (markenfreie) Netzmodell läßt sich in ein äquivalentes operationales (deklaratives) Objektmodell zurücktransformieren. Dazu brauchen lediglich die einzelnen Schritte der Transformationsmethode für operationale (deklarative) Objektmodelle in umgekehrter Richtung angewendet zu werden. Ein zweideutiger Sonderfall liegt nur dann vor, wenn ein markenfreies Netzmodell vorliegt. Denn dieses Netzmodell läßt sich sowohl als das Transformationsergebnis eines beliebigen deklarativen Objektmodells als auch das Transformationsergebnis eines speziellen operationalen Objektmodells auffassen. Der zweite Fall kann nicht ausgeschlossen werden, da operationale Netzmodelle zulässig sind, für deren Ausgangszustände überhaupt keine gültigen Vorkommnisse atomarer Grundtermformeln bekannt sind. Die Netzrepräsentation solcher operationalen Netzmodelle, deren Faktenmenge FAK_0 qua Voraussetzung leer ist, sind notwendig markenfreie operationale Netzmodelle. Vgl. zu den Transformationsmethoden, die in der voranstehenden Argumentation vorausgesetzt wurden, die anschließenden Erläuterungen. Dort wird auch aufgezeigt, daß die Transformation deklarativer (operationaler) Objektmodelle stets (in der Regel) zu markenfreien (markenbehafteten) Netzmodellen führt.

5.1.3.2.3 Transformation von Modellkonzeptualisierungen in Netzmodelle

5.1.3.2.3.1 Überblick

Es wird eine Methode vorgestellt, mit deren Hilfe sich ein Objektmodell in ein "äquivalentes"[1] Netzmodell transformieren läßt. Dabei wird vorausgesetzt, daß das Objektmodell denjenigen Realitätsausschnitt konzeptualisiert, der durch eine zugrundeliegende Modellierungsaufgabe als relevantes Modellierungsobjekt ausgezeichnet wurde. Darüber hinaus wird unterstellt, daß die ursprünglichen natürlichsprachlichen Objektkonzeptualisierungen und die daraus abgeleiteten prädikatenlogischen Objektmodelle alle Anforderungen erfüllen, die in den beiden vorangehenden Kapiteln dargelegt wurden. Folglich kann die Transformationsmethode von einem prädikatenlogischen Objektmodell ausgehen, das entweder in seiner Klausel- oder aber in seiner Produktionsregelform dargestellt ist. Entsprechend wird die Transformationsmethode in zwei Teilmethoden aufgespalten. Die eine ist auf die Umwandlung deklarativer Objektmodelle in Klauselform, die andere auf die Transformation von operationalen Objektmodellen in Produktionsregelform zugeschnitten.

Die Transformationsmethode für Objektmodelle in Klauselform wurde vom Verf. bereits an anderer Stelle für den einfacheren Fall von aussagenlogischen Objektmodellen vorgestellt und anhand eines umfangreicheren Beispiels aus dem Bereich der bilanzpolitischen Jahresabschlußgestaltung erläutert[2]. Sie wird nunmehr auf den erheblich größeren Ausdrucksreichtum der Prädikatenlogik 1. Stufe erweitert[3]. Eine ähnliche, aber weniger detaillierte Transformationsmethode für den prädikatenlogischen Modellierungsbereich haben auch MURATA und ZHANG[4] vorgelegt[5]. Eine eng verwandte Konstruktion von Netzmodellen findet sich bei LAUTENBACH[6], der jedoch seine Konstruktionsmethode nicht explizit spezifiziert[7]. Es existieren noch weitere Vorschläge, die sich mit der Abbildung logischer - vornehmlich prädikatenlogischer - Formelsysteme auf Netzmodelle befassen[8]. Doch sie unterscheiden sich entweder in ihrer konzeptionellen Vorgehensweise deutlich von dem hier vorgestellten Ansatz[9]. Oder sie fallen weitgehend mit dem zuvor angesprochenen Konzept von LAUTENBACH zusammen[10].

Transformationsmethoden für Objektmodelle in Produktionsregelform wurden dagegen bisher noch nicht systematisch ausgearbeitet. Zwar widmen sich mehrere Arbeiten intensiv der Repräsentation von Produktionsregelsystemen durch Netzmodelle. Hierzu gehören z.B.[11] die umfassende und subtil ausgearbeitete Studie von WEBER und WIEGAND[12] sowie die Beiträge von BAUMAN/TURANO[13] und VAGIN/ZAKHAROV/ROZENBLYUM[14]. Diesen Ansätzen fehlt jedoch eine Einbettung in den umfassenden prädikatenlogischen Kalkül. Statt dessen beschränken sie sich auf Produktionsregeldefinitionen, die jeweils nur eingeschränkte Varianten der oben vorgelegten allgemeinen Regeldefinition darstellen. Oftmals erreichen die Regelformalisierungen sogar nur das Ausdrucksniveau der Aussagenlogik[15]. Darüber hinaus wird zumeist auch nur exemplarisch argumentiert, ohne eine allgemeingültige Transformationsmethode anzugeben[16].

Schließlich leiden alle vorgenannten Beiträge zum Produktionsregel-Konzept unter dem Nachteil, daß sie sich an konventionellen Petrinetzen ausrichten, die keine Informationskanten und -stellen kennen. Dies hat zwei Konsequenzen. Erstens lassen sich die Regelvoraussetzungen von Produktionsregeln nicht mehr auf "natürliche" Weise modellieren. Denn die Formelvorkommnisse in den Regelvoraussetzungen, deren Gültigkeit durch die Regelausführungen nicht aufgehoben wird, besitzen die Qualität von Nebenbedingungen. Sie müssen durch 1-Schleifen repräsentiert werden[17]. Auf die Schwierigkeiten, Nebenbedingungen auf diese Weise im konventionellen Petrinetz-Konzept adäquat zu erfassen, wurde bereits ausführlich eingegangen[18]. Sie führen bei den o.a. Beiträgen dazu, daß Produktionsregeln in konventionellen Petrinetzen auf umständliche[19], mitunter sogar bemerkenswert intransparente[20] und problematische[21] Weise repräsentiert werden. Zweitens unterscheiden sich konventionelle Petrinetze, die keine Informationskanten und -stellen enthalten, prima facie[22)] kaum noch von den Netzmodellen für deklarative Objektmodelle, in denen Informationskanten und -stellen grundsätzlich nicht vorkommen[23)]. Infolgedessen wird die konzeptionelle Verschiedenartigkeit von operationalen und deklarativen Netzmodellen eher verwässert anstatt offengelegt.

Aufgrund aller vorgenannten Unzulänglichkeiten werden mit der anschließenden Präsentation von zwei Teilmethoden, die der Transformation von prädikatenlogischen Objektmodellen in Netzmodelle dienen, zwei Absichten verfolgt. Erstens soll durch die separate Transformation von deklarativen und operationalen Objektmodellen der grundsätzlichen Unterschiedlichkeit ihrer Klausel- bzw. Produktionsregelform Rechnung getragen werden. Zweitens soll den bereits vorliegenden systematischen Transformationsmethoden für deklarative Objektmodelle in Klauselform eine ähnlich systematische ausgearbeitete Methode zur Seite gestellt werden, welche die Transformation von operationalen Objektmodellen in Produktionsregelform leistet.

Anmerkungen zum Kapitel:

1) Bei der Transformation von Objektmodellen in Netzmodelle kann der Äquivalenzbegriff nicht in seiner strengen prädikatenlogischen Version verwendet werden. Denn die Konzeptualisierung des Modellierungsobjekts erfolgt über mehrere Konzeptualisierungsstufen hinweg, auf denen sich der Informationsgehalt des Objektmodells entsprechend zum jeweils erreichten Konzeptualisierungsstand verändert. Der prädikatenlogische Äquivalenzbegriff bedeutet dagegen die Invarianz der logischen Konsequenzenmenge eines Formelsystems. Der Informationsgehalt eines Objektmodells läßt sich mit dieser Konsequenzenmenge identifizieren. Dann schließen sich die Variabilität des Informationsgehalt und die Unveränderlichkeit der Konsequenzenmenge gegenseitig aus.
Eine strenge prädikatenlogische Äquivalenz besteht nur zwischen den prädikatenlogischen Objektmodellen in Klausel- oder Produktionsregelform einerseits und den Netzmodellen in Klausel- bzw. Produktionsregelform andererseits. Diese Modelläquivalenz im strengen Sinne wird zum Anlaß genommen, um eine Äquivalenz zwischen Modellkonzeptualisierungen und Netzmodellen in abgeschwächter Weise einzuführen: Eine Modellkonzeptualisierung und ein daraus abgeleitetes Netzmodell heißen genau dann (schwach) äquivalent, wenn sich die Modellkonzeptualisierung so in ein prädikatenlogisches Objektmodell in Klausel- oder Produktionsregelform umwandeln läßt, daß dieses Objektmodell mit dem Netzmodell im prädikatenlogischen Sinne äquivalent ist.
Die hierbei vorausgesetzte prädikatenlogische Äquivalenz zwischen einem prädikatenlogischen Objektmodell in Klausel- oder Produktionsregelform einerseits und einem Netzmodell andererseits wird in dieser Arbeit nicht bewiesen. Sie läßt sich aber aus den beiden Teilmethoden ableiten, die für die Transformation solcher Objekt- in Netzmodelle vorgelegt werden. Vgl. zur prädikatenlogischen Äquivalenz von Objektmodellen in Klauselform und den daraus abgeleiteten Netzmodellen auch MURATA,TA. (1988b), S. 486 u. 495, und ZHANG,D. (1989).

2) Vgl. ZELEWSKI (1986b), S. 11ff. u. 35ff.; ZELEWSKI (1989c), S. 17ff. u. 47ff.; ZELEWSKI (1989e), S. 69ff. u. 77ff.

3) Vgl. dazu auch den entsprechenden Hinweis bei ZELEWSKI (1989c), S. 24.

4) Vgl. MURATA,TA. (1988b), S. 484f.; vgl. darüber hinaus auch MURATA,TA. (1988a), S. 82ff., und ZHANG,D. (1989).

5) Die beiden Autoren befassen sich nur mit der Transformation von HORN-Klauseln. Diese Einschränkung ist jedoch nicht erforderlich. Denn der Verf. wird nachfolgend zeigen, daß sich beliebige allgemeine Klauseln in äquivalente Teilnetze überführen lassen.

6) Vgl. LAUTENBACH (1985b), S. 4ff., insbesondere S. 7ff. u. 29f.

7) Außerdem läßt LAUTENBACH im Gegensatz zu MURATA und ZHANG die Möglichkeit von Transitionsbeschriftungen außer acht.

8) Vgl. THIELER-MEVISSEN (1975), S. 2ff.; THIELER-MEVISSEN (1977), S. 6ff. u. 33ff.; MAINZ (1984), S. 11ff.; FIDELAK (1986a), S. 19ff.; FIDELAK (1986b), S. 34ff.; FIDELAK (1987a), S. 95ff.; LIU,N. (1987), S. 121ff.; OBERWEIS (1988b), S. 300ff.; VAGIN (1988), S. 100f.

9) Dies gilt vor allem für die Arbeiten von THIELER-MEVISSEN, die auf Faktnetzen beruhen, und von VAGIN, ZAKHAROV und ROZENBLYUM, die sich stark an Produktionsregelsysteme anlehnen (darauf wird in Kürze eingegangen).

10) Das trifft vor allem auf die Ausführungen von MAINZ und FIDELAK zu.

11) An anderer Stelle wird ausführlicher auf die Repräsentation von Produktionsregeln durch Netzmodelle eingegangen.

12) Vgl. WEBER,A. (1987), S. 162ff., insbesondere S.180ff.

13) Vgl. BAUMAN (1986), S. 193f.

14) Vgl. VAGIN (1988), S. 100f. VAGIN, ZAKHAROV und ROZENBLYUM beziehen sich zwar explizit nur auf die Transformation von aussagen- und prädikatenlogischen Formelsystemen in Netzmodelle. Aber ihre Ausführungen lassen sich so auslegen, daß zustandsverändernde Produktionsregeln erfaßt werden. Denn die dort konstruierten Netze werden in einer Weise benutzt, die der operationalen Semantik von Produktionsregeln gleicht. Darauf verweisen drei Indizien. Erstens werden die Formeln, die den Ausgangsstellen einer Transition zugeordnet sind, miteinander *konjunktiv* verknüpft; vgl. Fall 2b bei VAGIN (1988), S. 100f. Dies wird an anderer Stelle als Charakteristikum von Produktionsregeln gegenüber der adjunktiven Verknüpfung der Konklusionsformeln aus gewöhnlichen Subjugaten herausgestellt. Zweitens werden keine markenfreien Netze verwendet, wie es für deklarative Netzmodelle in Klauselform korrekt wäre. Statt dessen werden die Netze so markiert, wie es für die Repräsentation von Produktionsregeln in operationalen Netzmodellen typisch ist. Drittens ähnelt die Notation der Formelsysteme, die bei VAGIN (1988), S. 100ff., den modellierten Netzen zugrundeliegen, der typischen Darstellungsweise von Produktionsregelsystemen. Aufgrund der vorgenannten Indizien wird auf die Arbeit VAGIN (1988), stets unter der interpre-

tierenden Prämisse Bezug genommen, daß sie tatsächlich Netzrepräsentationen auf der Grundlage des Produktionsregel-Konzepts intendiert haben.

15) Dies ist beispielsweise für die Beiträge von BAUMAN (1986), S. 193f., und VAGIN (1988), S. 100f., der Fall.

16) Z.B. wird in VAGIN (1988), S. 100f., keine allgemeingültige Methode expliziert, mit deren Hilfe sich jedes (prädikaten)logische Formelsystem in ein äquivalentes Netzmodell transformieren ließe. Die Ausführungen bleiben statt dessen im Exemplarischen verhaftet.

17) Vgl. zur Verwendung von 1-Schleifen bei der Repräsentation von Produktionsregeln, um die unveränderte Gültigkeit ihrer Regelvoraussetzungen zu modellieren, BAUMAN (1986), S. 193f.; VAGIN (1988), S. 101, Fig. 2d.

18) 1-Schleifen, die zur Repräsentation der Nebenbedingungen (Regelvoraussetzungen) von Produktionsregeln erforderlich sind, arbeiten mit dem Mittel der Fehlerkompensation. Denn sie beruhen auf dem Modellierungsfehler, die Gültigkeit einer Nebenbedingung durch das Schalten der bedingungszugehörigen Transition zunächst dadurch aufzuheben, daß von der bedingungsrepräsentierenden Stelle eine Marke abgezogen wird. Dies widerspricht dem Charakter von Nebenbedingungen - und hier speziell: von Regelvoraussetzungen -, daß ihre Gültigkeit durch die Ausführung der bedingten Aktion - der Produktionsregel - nicht beeinflußt wird. Dieser konzeptionelle Fehler wird bei 1-Schleifen nachträglich wieder aufgehoben, indem eine Marke auf die bedingungsrepräsentierende Stelle zurückgelegt wird. Vgl. dazu die detailliertere Erläuterung der Problematik von Nebenbedingungen und 1-Schleifen.

19) Die Umständlichkeit folgt schon aus dem Mechanismus der Fehlerkompensation von 1-Schleifen. Darüber hinaus kann der Verzicht auf Informationskanten und -stellen zu weiteren Komplizierungen führen, deren detaillierte Kritik jedoch nicht mehr im Erkenntnisinteresse dieser Arbeit liegt. Vgl. als Beleg etwa die aufwendigen, aber keineswegs notwendigen Netzkonstrukte bei VAGIN (1988), S. 101, Fig. 2d, sowie S. 102, Fig. 4. Sie wären nicht erforderlich, wenn die u.a. Transformationsmethode für operationale Objektmodelle angewendet worden wäre.

20) Vgl. als Beispiele intransparenter Netzkonstruktionen VAGIN (1988), S. 101, Fig. 2d, sowie S. 102, Fig. 4.

21) Ein erster problematischer Aspekt liegt bereits in der Fehlerkompensation, die kurz zuvor angesprochen wurde. Hinzu kommt jedoch, daß die Technik der 1-Schleifen - trotz aller konzeptionellen Vorbehalte - mitunter noch nicht einmal konsequent durchgehalten wird. Z.B. verwenden VAGIN, ZAKHAROV und ROZENBLYUM in VAGIN (1988), S. 101, einmal eine 1-Schleife, um die unveränderte Gültigkeit einer Regelvoraussetzung auszudrücken (Fig. 2d), unterlassen jedoch deren Verwendung im selben Kontext für analoge Regeln (Fig. 2a u. 2b). Es läßt sich überhaupt nicht nachvollziehen, warum einmal garantiert wird, die Gültigkeit der Regelvoraussetzung im Netzmodell sicherzustellen, die beiden andere Male auf eben diese Garantie aber zu verzichten.

22) Bei näherem Hinsehen verbleiben noch Differenzen, welche die Verknüpfungslogik der Ausgangsstellen von Transitionen und die Netzmarkierungen betreffen. Ausführlich wird auf die Unterschiede zwischen Netzmodellen, die entweder für deklarative Objektmodelle in Klauselform oder aber für operationale Objektmodelle in Produktionsregelform definiert sind, an anderer Stelle eingegangen.

23) Vgl. dazu die nachfolgende Transformationsmethode für deklarative Netzmodelle.

5.1.3.2.3.2 Transformation deklarativer Objektmodelle

Methode zur Transformation von deklarativen Objektmodellen in Netzmodelle:

a) Die Transformationsmethode startet mit einem deklarativen Objektmodell in der Klauselform $I_0(KL_{FS})$.

b) Das Definitionstupel $SN_{FS} = (TOP,SPEC_{MSIG};BES,M_0;IB)$ eines Synthetischen Netzes wird für den Ausgangszustand "r=0" des Netzmodells durch folgende Festlegungen initialisiert:

❑ Die Stellen-, Transitionen- und Kantenmenge aus der Netztopologie TOP=(S,T;F) sind jeweils leer: S := ∅, T := ∅ und F := ∅.

❑ Die Netzspezifikation $SPEC_{MSIG} = (SO,OP,PRÄ;OBF,OPF,FAK_0;TTMF(VAF),VBM;RES;TR;ÜS)$ läßt sich weitgehend aus dem prädikatenlogischen Objektmodell ableiten:

- Die Sortenmenge SO umfaßt alle Sorten $sort_i$, die im prädikatenlogischen Objektmodell vorkommen. Dabei kann es sich jeweils um Attribute $attribut_q$ für die Repräsentation von Objekteigenschaften oder um sortierte Marken sor_marke_s für die Darstellung von Objektklassen handeln.
- Die Menge OP enthält alle Operationssymbole Op_j, die im Objektmodell dazu dienen, komplexe Objekteigenschaften oder Objektklassen aus einfacheren Eigenschaften bzw. Klassen zusammenzusetzen.
- Die Menge PRÄ führt alle Prädikatssymbole $Prä_u$ auf, aus denen im Objektmodell prädikatenlogische Formeln $prä_u$ abgeleitet sind. Die Formelvorkommnisse drücken jeweils zustandsbeschreibende Aussagen, Ausprägungen von Objekteigenschaften oder Objektbeziehungen aus.
- Die Familie OBF ergibt sich aus den Mengen OB_i formaler Objekte, die im Objektmodell für alle Sorten $sort_i$ festliegen. Die Objektmengen OB_i definieren alle zulässigen Eigenschaftsausprägungen und alle zulässigen Objekte.
- Die Familie OPF umfaßt alle Operationen op_j, die aus Operationssymbolen Op_j abgeleitet sind. Sie dienen im Objektmodell dazu, die Ausprägungen komplexer Objekteigenschaften oder komplexe Objekte aus einfacheren Eigenschaftsausprägungen bzw. Objekten zusammenzusetzen.
- Die Familie FAK_0 enthält für jedes Prädikatssymbol $Prä_u$ aus der Menge PRÄ die leere Faktenmenge: $FAK_{u.0} := \emptyset$ für alle $u \in \{1,...,U\}$.
- Die Familie VAF enthält alle sortenspezifischen Variablenmengen VA_i, deren Elemente Variablen aus den Formelvorkommnissen des Objektmodells darstellen.
- Für die Familie TTMF(VAF) teilevaluierter Terme und für die Menge VBM aller Variablenbindungsfunktionen werden ihre Definitionen aus der Darstellung Synthetischer Netze unverändert übernommen.
- Die Restriktionenmenge RES wird auf die Menge $FOR_{standard}$ aller algebraischen Standardformeln festgesetzt: $RES := FOR_{standard}$.
- Die Transaktionenmenge TR wird als leere Menge initialisiert: TR := ∅.

- Das allgemeine Übergangsschema ÜS wird wiederum aus der Definition Synthetischer Netze übernommen.

❑ Die Abbildungsvorschriften der Beschriftungsfunktionen bsp, bsk, btt und bfm aus die Netzbeschriftung BES = {bsp,bsk,btt,bfm} werden durch binäre Anschriftenmengen[1] BSP, BSK, BTT bzw. BFM dargestellt und jeweils durch die leere Menge eingeführt: BSP := ∅, BSK := ∅, BTT := ∅ und BFM := ∅.

❑ Die Ausgangsmarkierung M_0 bildet die Stellenmenge S auf die Menge ab, die nur die leere Faktenmenge ∅ enthält: M_0: S→{∅}[2].

❑ Die Menge IB der Integritätsbedingungen wird aus der Definition Synthetischer Netze unverändert übernommen.

c) Die nachfolgende Operationenfolge wird für jede Klausel kla_p aus der Klauselmenge KL_{FS} mit p∈{1,...,P} und für jedes klauselzugehörige Literal $lit_{p.q}$ aus der Literalmenge LIT_p mit q∈{1,...,Q_p} wiederholt[3]:

❑ Für das Literal $lit_{p.q}$ wird genau eine literalrepräsentierende Stelle $s_{m(p.q)}$ so eingeführt, daß ihr Index "m(p.q)" das Literal $lit_{p.q}$ eindeutig identifiziert[4].

❑ Die Stelle $s_{m(p.q)}$ wird mit dem Namen $Prä_u$ desjenigen Prädikatssymbols $Prä_u(sor\_marke_{s(u.1)} \ldots sor\_marke_{s(u.Ku)})$ beschriftet, aus dem das Literal $lit_{p.q}$ abgeleitet wurde:

$$(lit_{p.q} :\Leftrightarrow prä_{u(p.q)}(te_1,\ldots,te_{Ku})) \vee (lit_{p.q} :\Leftrightarrow \neg prä_{u(p.q)}(te_1,\ldots,te_{Ku}))$$
$$\rightarrow \; bsp(s_{m(p.q)}) = Prä_u$$

d) Die anschließende Operationenfolge wird für alle Klauseln kla_p aus der Klauselmenge KL_{FS} mit p∈{1,...,P} ausgeführt[5]:

❑ Für die Klausel kla_p wird genau eine klauselrepräsentierende Transition $t_{n(p)}$ eingeführt.

❑ Die topologische Umgebung der Transition $t_{n(p)}$ wird durch die leeren Mengen $VB(t_{n(p)}) := \emptyset$ und $NB(t_{n(p)}) := \emptyset$ initialisiert.

❑ Die Transitionenmenge T wird um die Transition $t_{n(p)}$ erweitert:

$$T := T \cup \{t_{n(p)}\}.$$

❑ Die Transition $t_{n(p)}$ wird mit dem identisch indizierten Transaktionsnamen $tr_{n(p)}$ beschriftet: $btt(t_{n(p)}) = tr_{n(p)}$.

❑ Die Anschriftenmenge BTT für die Abbildungsvorschrift der Beschriftungsfunktion btt wird um die Beschriftung der Transition $t_{n(p)}$ mit dem Transaktionsnamen $tr_{n(p)}$ erweitert:

$$BTT := BTT \cup \{(t_{n(p)}, tr_{n(p)})\}$$

❑ Die Definition der Transaktion $tr_{n(p)}$ wird mit den leeren Mengen $VB(tr_{n(p)}) := \emptyset$, $MTAV_{n(p)} := \emptyset$, $IB(tr_{n(p)}) := \emptyset$, $MTAI_{n(p)} := \emptyset$, $NB(tr_{n(p)}) := \emptyset$, $MTAN_{n(p)} := \emptyset$ und $RES_{n(p)} := \emptyset$ initialisiert.

❑ Die Transition $t_{n(p)}$, welche die Klausel kla_p repräsentiert, wird mit der Stelle $s_{m(p.q)}$, die das Literal $lit_{p.q}$ darstellt, vermittels einer Ausgangskante $(t_{n(p)}, s_{m(p.q)})$ verknüpft, falls die Literalmenge LIT_p der Klausel kla_p das positive Literal $prä_{u(p.q)}$ umfaßt:

$$prä_{u(p.q)} \in LIT_p \;\rightarrow\; ka_{n(p).m(p.q)} = (t_{n(p)}, s_{m(p.q)})$$

- Die Kantenmenge (Flußrelation) F wird um jede Ausgangskante der Transition $t_{n(p)}$ erweitert:

  $prä_{u(p.q)} \in LIT_p \rightarrow F := F \cup \{(t_{n(p)}, s_{m(p.q)})\})$

- Der Nachbereich $NB(t_{n(p)})$ der Transition $t_{n(p)}$ wird um jede Stelle $s_{m(p.q)}$ erweitert, mit der die Transition $t_{n(p)}$ durch eine Ausgangskante verbunden ist:

  $NB(t_{n(p)}) := NB(t_{n(p)}) \cup \{s_{m(p.q)}: (t_{n(p)}, s_{m(p.q)}) \in F\}$

- Der Nachbereich $NB(tr_{n(p)})$ der Transaktion $tr_{n(p)}$ wird um jedes Prädikatssymbol $Prä_u$ erweitert, aus dem ein Formelvorkommnis $prä_{u(p.q)}$ abgeleitet ist, das die Klausel kla_p der Transition $t_{n(p)}$ in ihrer Literalmenge LIT_p als positives Literal umfaßt:

  $NB(tr_{n(p)}) := NB(tr_{n(p)}) \cup \{Prä_u: Prä_u \in PRÄ \wedge prä_{u(p.q)} \in LIT_p\}$

- Jede Ausgangskante $(t_{n(p)}, s_{m(p.q)})$ der Transition $t_{n(p)}$ wird mit der "Multimenge" beschriftet, die aus dem genau einen[6)] atomaren Formelvorkommnis $prä_{u(p.q)}$ besteht, das als positives Literal $lit_{p.q} :\Leftrightarrow prä_{u(p.q)}(te_1, \ldots, te_{Ku})$ in der Literalmenge LIT_p der Klausel kla_p enthalten ist und für dessen Repräsentation die Ausgangskante eingeführt wurde:

  $MTAN_{u(p.q).n(p)} = \{(1, prä_{u(p.q)}(te_1, \ldots, te_{Ku}))\}$

  $prä_{u(p.q)} \in LIT_p \rightarrow bfm(t_{n(p)}, s_{m(p.q)}) = MTAN_{u(p.q).n(p)}$

- Die Anschriftenmenge BFM für die Abbildungsvorschrift der Beschriftungsfunktion bfm wird um die Beschriftungen aller Ausgangskanten der Transition $t_{n(p)}$ erweitert:

  $bfm(t_{n(p)}, s_{m(p.q)}) = MTAN_{u(p.q).n(p)}$
  $\rightarrow BFM := BFM \cup \{((t_{n(p)}, s_{m(p.q)}), MTAN_{u(p.q).n(p)})\}.$

- Für den gesamten Nachbereich der Transaktion $tr_{n(p)}$ gilt:

  $MTAN_{n(p)} := \{MTAN_{u(p.q).n(p)}: MTAN_{u(p.q).n(p)} = bfm(t_{n(p)}, s_{m(p.q)}) \wedge \ldots$
  $bsp(s_{m(p.q)}) = Prä_u \wedge Prä_u \in NB(tr_{n(p)}) \wedge btt(t_{n(p)}) = tr_{n(p)}\}$

- Die Transition $t_{n(p)}$, welche die Klausel kla_p repräsentiert, wird mit der Stelle $s_{m(p.q)}$, die das Literal $lit_{p.q}$ darstellt, vermittels einer Eingangskante $(s_{m(p.q)}, t_{n(p)})$ verknüpft, falls die Literalmenge LIT_p der Klausel kla_p das negative Literal $\neg prä_{u(p.q)}$ umfaßt:

  $\neg prä_{u(p.q)} \in LIT_p \rightarrow ka_{m(p.q).n(p)} = (s_{m(p.q)}, t_{n(p)})$

- Die Kantenmenge (Flußrelation) F wird um jede Eingangskante der Transition $t_{n(p)}$ erweitert:

  $\neg prä_{u(p.q)} \in LIT_p \rightarrow F := F \cup \{(s_{m(p.q)}, t_{n(p)})\})$

- Der Vorbereich $VB(t_{n(p)})$ der Transition $t_{n(p)}$ wird um jede Stelle $s_{m(p.q)}$ erweitert, mit der die Transition $t_{n(p)}$ durch eine Eingangskante verbunden ist:

  $VB(t_{n(p)}) := VB(t_{n(p)}) \cup \{s_{m(p.q)}: (s_{m(p.q)}, t_{n(p)}) \in F\}$

- Der Vorbereich $VB(tr_{n(p)})$ der Transaktion $tr_{n(p)}$ wird um jedes Prädikatssymbol $Prä_u$ erweitert, aus dem ein negatives Literal $\neg prä_{u(p.q)}$ abgeleitet ist, das die Klausel kla_p der Transition $t_{n(p)}$ in ihrer Literalmenge LIT_p umfaßt:

  $$VB(tr_{n(p)}) := VB(tr_{n(p)}) \cup \{Prä_u\text{: } Prä_u \in PRÄ \wedge (\neg prä_{u(p.q)} \in LIT_p)\}$$

- Jede Eingangskante $(s_{m(p.q)}, t_{n(p)})$ der Transition $t_{n(p)}$ wird mit der "Multimenge" beschriftet, die aus dem genau einen atomaren Formelvorkommnis $prä_{u(p.q)}$ besteht, das als negatives Literal $lit_{p.q}{:}\Leftrightarrow\neg prä_{u(p.q)}(te_1,...,te_{Ku})$ in der Literalmenge LIT_p der Klausel kla_p enthalten ist und für dessen Repräsentation die Eingangskante eingeführt wurde:

  $$MTAV_{u(p.q).n(p)} = \{(1, prä_{u(p.q)}(te_1,...,te_{Ku}))\}$$

  $$\neg prä_{u(p.q)} \in LIT_p \;\rightarrow\; bfm(s_{m(p.q)}, t_{n(p)}) = MTAV_{u(p.q).n(p)}$$

- Die Anschriftenmenge BFM für die Abbildungsvorschrift der Beschriftungsfunktion bfm wird um die Beschriftungen aller Eingangskanten der Transition $t_{n(p)}$ erweitert:

  $$bfm(s_{m(p.q)}, t_{n(p)}) = MTAV_{u(p.q).n(p)}$$

  $$\rightarrow\; BFM := BFM \cup \{((s_{m(p.q)}, t_{n(p)}), MTAV_{u(p.q).n(p)})\}$$

- Für den gesamten Vorbereich der Transaktion $tr_{n(p)}$ gilt:

  $$MTAV_{n(p)} := \{MTAV_{u(p.q).n(p)}\text{: } MTAV_{u(p.q).n(p)} = bfm(s_{m(p.q)}, t_{n(p)}) \wedge ...$$
  $$bsp(s_{m(p.q)}) = Prä_u \wedge Prä_u \in VB(tr_{n(p)}) \wedge btt(t_{n(p)}) = tr_{n(p)}\}$$

- Die Transaktionenmenge TR des Synthetischen Netzes wird um die Transaktion $tr_{n(p)}$ erweitert:

  $$TR := TR \cup \{tr_{n(p)}\}$$

e) Alle literalrepräsentierenden Stellen $s_{m(p.q)}$, die zwar für verschiedene[7)] atomare oder negierte Formelvorkommnisse $prä_{u(p.q)}$ bzw. $\neg prä_{u(p.q)}$ eingeführt wurden, aber jeweils mit dem Namen $Prä_u$ desselben Prädikatssymbols beschriftet sind, werden miteinander identifiziert und durch genau eine Stelle ersetzt. Danach liegt für jedes Prädikatssymbol genau eine Stelle vor, die mit dem Namen des Prädikatssymbols beschriftet ist. Die Indizierung aller Stellen wird derart erneuert, daß die neuen Indizes "m" die Werte $m=1,...,M$ lückenlos durchlaufen. Die Stellenmenge S des Netzes ist dann die Menge aller reindizierten Stellen s_m.

f) Alle voranstehend eingeführten Netzkonstrukte werden an die Stellenidentifizierung und Reindexierung angepaßt. Zu diesem Zweck werden alle mehrfachen Vorkommnisse derselben Stelle s_m durch genau ein Stellenvorkommnis s_m ersetzt. Die Stellenmenge S des Netzes ist dann die Menge aller Stellen s_m, die nach der Stellenidentifizierung in jeweils noch genau einem Vorkommnis vorhanden sind.

g) Die ursprünglich leere Anschriftenmenge BSP für die Abbildungsvorschrift der Beschriftungsfunktion bsp wird um die Beschriftungen aller Stellen s_m aus der Stellenmenge S mit Prädikatssymbolen $Prä_u$ erweitert, die nach der Stellenidentifizierung und Reindexierung vorliegen:

$$bsp(s_m) = Prä_u \;\rightarrow\; BSP := BSP \cup \{(s_m, Prä_u)\}$$

h) Für jede Stelle s_m aus der Stellenmenge S, der mittels der Beschriftungsfunktion $bsp(s_m) = Prä_u$ das Prädikatssymbol $Prä_u$ zugeordnet ist, wird festgelegt, wie viele gültige Formelvorkommnisse $prä_u$ unter jeder beliebigen Interpretation I_r des prädikatenlogischen Objektmodells maximal existieren dürfen[8]. Diese obere Schranke der Kardinalität der prädikatszugehörigen Faktenmenge $FAK_{u.r}$ ist die Markenkapazität KAP_m der Stelle s_m mit $KAP_m \in \mathcal{N}_+$. Die Stelle s_m wird mit ihrer Markenkapazität beschriftet: $bsk(s_m) = KAP_m$.

i) Die ursprünglich leere Anschriftenmenge BSK für die Abbildungsvorschrift der Beschriftungsfunktion bsk wird um die Beschriftungen aller Stellen s_m aus der Stellenmenge S mit ihren Markenkapazitäten KAP_m erweitert:

$$bsk(s_m) = KAP_m \rightarrow BSK := BSK \cup \{(s_m, KAP_m)\}$$

j) Jeder Stelle s_m wird durch die Ausgangsmarkierung M_0 die leere Faktenmenge $FAK_{u.0} = \emptyset$ desjenigen Prädikatssymbols zugeordnet, mit dessen Namen $Prä_u$ sie beschriftet wurde. Für alle $m \in \{1,...,M\}$ gilt also:

$$bsp(s_m) = Prä_u \rightarrow M_0(s_m) := FAK_{u.0} = \emptyset$$

k) Falls in der Literalmenge LIT_p einer Klausel kla_p mehrere positive Literale $lit_{p.q}{:}\Leftrightarrow prä_{u(p.q)}$ oder mehrere negative Literale $lit_{p.q}{:}\Leftrightarrow \neg prä_{u(p.q)}$ verschiedene Vorkommnisse desselben Prädikatssymbols $Prä_u$ sind, gilt nach der Stellenidentifizierung, die im voranstehenden Transformationsschritt erfolgte: Die Transition $t_{n(p)}$, welche die Klausel kla_p repräsentiert, wird mit der Stelle s_m, die wegen $bsp(s_m) = Prä_u$ das Prädikatssymbol $Prä_u$ repräsentiert, durch ein Bündel aus mehreren Aus- bzw. Eingangskanten verknüpft[9]. Diese Aus- und Eingangskanten sind mit den literalspezifischen "Multimengen" $MTAN_{u(p.q).n(p)}$ bzw. $MTAV_{u(p.q).n(p)}$ beschriftet. Jede dieser "Multimengen" besteht aus dem literalspezifischen atomaren Formelvorkommnis $prä_{u(p.q)}(te_{1.d},..., te_{Ku.d})$[10] in der Multiplizität $mu_{u(p.q)} = 1$[11]. Analog zur vorangehenden Stellenidentifizierung lassen sich auch alle gleichgerichteten Kanten aus einem Kantenbündel zu einer Multikante zusammenfassen[12]. Diese Multikante wird nunmehr mit der Multimenge - oder äquivalent: mit der formalen Summe[13] - der atomaren Formelvorkommnisse $prä_{u(p.q)}(te_{1.d},...,te_{Ku.d})$ aller Literale $lit_{p.q}$ beschriftet, die aus demselben Prädikatssymbol $Prä_u$ abgeleitet wurden. Für die Zusammenfassung von zwei Eingangskanten, welche die Transition $t_{n(p)}$ mit derselben Stelle s_m verknüpfen, zu einer Multikante gilt:

Vorausgesetzt werden:

$$
\begin{aligned}
& \mathrm{bsp}(s_{m(p.q1)}) = \mathrm{bsp}(s_{m(p.q2)}) = \mathrm{Prä}_u \\
\wedge\ & s_{m(p.q1)} \in VB(t_{n(p)}) \ \wedge\ s_{m(p.q2)} \in VB(t_{n(p)}) \\
\wedge\ & \mathrm{bfm}(s_{m(p.q1)}, t_{n(p)}) = MTAV_{u(p.q1).n(p)} \\
& \qquad = \{1, \mathrm{prä}_{u(p.q1)}(te_{1.d1},\ldots,te_{Ku.d1})\} \\
& \qquad = 1 \bullet \mathrm{prä}_{u(p.q1)}(te_{1.d1},\ldots,te_{Ku.d1}) \\
& \qquad = \mathrm{prä}_{u(p.q1)}(te_{1.d1},\ldots,te_{Ku.d1}) \\
\wedge\ & \mathrm{bfm}(s_{m(p.q2)}, t_{n(p)}) = MTAV_{u(p.q2).n(p)} \\
& \qquad = \{1, \mathrm{prä}_{u(p.q2)}(te_{1.d2},\ldots,te_{Ku.d2})\} \\
& \qquad = 1 \bullet \mathrm{prä}_{u(p.q2)}(te_{1.d2},\ldots,te_{Ku.d2}) \\
& \qquad = \mathrm{prä}_{u(p.q2)}(te_{1.d2},\ldots,te_{Ku.d2})
\end{aligned}
$$

Durch Stellenidentifizierung $s_{m(p.q1)} = s_{m(p.q2)}$ und Reindexierung $m(p.q_1) = m(p.q_2) = m$ folgt daraus:

$$
\begin{aligned}
& \mathrm{bsp}(s_m) = \mathrm{Prä}_u \ \wedge\ s_m \in VB(t_{n(p)}) \\
\wedge\ & \mathrm{bfm}(s_m, t_{n(p)}) = MTAV_{u.n(p)} \\
& \qquad = \{(1, \mathrm{prä}_u(te_{1.d1},\ldots,te_{Ku.d1})), (1, \mathrm{prä}_u(te_{1.d2},\ldots,te_{Ku.d2}))\} \\
& \qquad = 1 \bullet \mathrm{prä}_u(te_{1.d1},\ldots,te_{Ku.d1}) + 1 \bullet \mathrm{prä}_u(te_{1.d2},\ldots,te_{Ku.d2}) \\
& \qquad = \mathrm{prä}_u(te_{1.d1},\ldots,te_{Ku.d1}) + \mathrm{prä}_u(te_{1.d2},\ldots,te_{Ku.d2})
\end{aligned}
$$

Die Zusammenfassung zweier Ausgangskanten zu einer Multikante erfolgt analog. Durch sukzessives Wiederholen der voranstehenden Zusammenfassungsoperation für je zwei Kanten läßt sich jedes Kantenbündel auf genau eine Multikante reduzieren. Diese Reduktionsoperation wird auf alle Kanten des Netzes angewandt. Auf diese Weise wird erreicht, daß die graphische Netzrepräsentation niemals einen Multi-, sondern stets einen Monographen darstellt[14)].

l) Das Synthetische Netz $SN_{FS} = (TOP, SPEC_{MSIG}; BES, M_0; IB)$, das nach allen voranstehenden Transformationsschritten vorliegt, ist das gesuchte deklarative Netzmodell für das Objektmodell in Klauselform $I_0(KL_{FS})$.

<u>Erläuterungen und Ergänzungen zur Transformationsmethode:</u>

a) Bei jedem deklarativen Netzmodell handelt es sich um ein markiertes, aber markenfreies Netz, weil jeder Stelle s_m die leere Ausgangsmarkierung $M_0(s_m) = \emptyset$ zugeordnet wird.

b) In einem deklarativen Netzmodell entspricht jede Transition genau einer Klausel aus dem zugrundeliegenden prädikatenlogischen Objektmodell. Wegen dieser bijektiven Zuordnungsmöglichkeit zwischen Transitionen und Klauseln wird auch von klauselspezifischen, -zugehörigen oder -repräsentierenden[15)] Transitionen gesprochen.

c) Jede Stelle gibt genau ein Prädikatssymbol aus dem Objektmodell wieder. Sie wird daher auch als prädikatspezifische, -zugehörige oder -repräsentierende Stelle bezeichnet.

d) Zunächst wird jede Klausel kla_p aus der Formel $prä_{FS}$, die den Ausgangszustand des prädikatenlogischen Objektmodells in konjunktiver Normalform beschreibt, durch genau eine Transition mit inzidenten Ein- und Ausgangsstellen abgebildet. Diese Klauselrepräsentationen erfolgen voneinander unabhängig, so daß die resultierenden Teilnetze zunächst unverknüpft nebeneinanderstehen. Vermittels der Stellenidentifizierung werden aber literalrepräsentierende Stellen zusammengelegt, falls sie jeweils mit demselben Prädikatssymbol $Prä_u$ beschriftet sind. In der Regel kommen in prädikatenlogischen Objektmodellen Vorkommnisse desselben Prädikatssymbols in verschiedenen Klauseln vor. Für diesen Regelfall bedeutet die Stellenidentifizierung, daß die zunächst isolierten Teilnetze über gemeinsame Stellen miteinander verknüpft werden. Dieser stelleninduzierte Netzzusammenhang gibt die konjunktive Verknüpfung aller Klauseln aus der zustandsbeschreibenden Formel $prä_{FS}$ im Netzmodell wieder. Daher stellen Netzmodelle in der Regel im zusammenhängende Netze dar.

e) Es kann allerdings der Sonderfall eintreten, daß alle Literale einer Klausel aus Prädikatssymbolen abgeleitet worden sind, die keinem einzigen Literal in irgendeiner anderen Klausel zugrundeliegen. Dann ist das Teilnetz aus der Transition, welche die erstgenannte Klausel repräsentiert, und aus ihren literalrepräsentierenden inzidenten Stellen von allen anderen Teilnetzen isoliert. In diesem Sonderfall stellt das Netzmodell ein unzusammenhängendes Netz dar.

f) Wenn von den technischen Details der o.a. Transformationsmethode abgesehen wird, entspricht jeder Klausel kla_p, die aus der konjunktiven Normalform $prä_{FS}$ eines Objektmodells stammt, im Netzmodell genau ein Teilnetz mit der klauselspezifischen Topologie $TOP(kla_p)$. Hauptbestandteil der Teilnetztopologie ist die eine klauselrepräsentierende Transition $t_{n(p)}$. Diese Transition besitzt für jedes negative Literal der Klausel genau eine Eingangsstelle und für jedes positive Literal der Klausel genau eine Ausgangsstelle. Daher ist die klauselspezifische Netztopologie $TOP(kla_p)$ für eine Klausel kla_p mit "$\approx$" als Notation für eine bijektive Zuordnung definiert durch:

$$TOP(kla_p) = (S_p,T_p;F_p) \quad \approx \quad kla_p :\Leftrightarrow lit_{p.1} \wedge \ldots \wedge lit_{p.Qp}$$

mit:

- ❑ $S_p = \{s_{m(p.q)}: q=1,\ldots,Q_p\}$
- ❑ $T_p = \{t_{n(p)}\}$
- ❑ $F_p = \{ka_{p.q}: q=1,\ldots,Q_p\}$
- ❑ und für alle $q \in \{1,..,Q_p\}$:

$$ka_{p.q} = \begin{cases} (s_{m(p.q)},t_{n(p)}); & \text{falls } lit_{p.q} :\Leftrightarrow \neg prä_{u(p.q)} \wedge bsp(s_{m(p.q)}) = Prä_u \\ (t_{n(p)},s_{m(p.q)}); & \text{falls } lit_{p.q} :\Leftrightarrow prä_{u(p.q)} \wedge bsp(s_{m(p.q)}) = Prä_u \end{cases}$$

mit beliebigen atomaren Formelvorkommnissen $prä_{u(p.q)}$

g) Die Transformation einer Klausel kla_p in ein klauselspezifisches Teilnetz läßt sich besonders anschaulich charakterisieren, wenn die Klausel in subjunktiver Weise dargestellt ist. Dann entspricht jedem atomaren Formelvorkommnis $prä_{u(p.q)}$ aus dem Antezedens der Subjugatformel kla_p eine Eingangsstelle der klauselrepräsentierenden Transition $t_{n(p)}$ und jedem atomaren Formelvorkommnis $prä_{u(p.q)}$ aus der Konklusion der Subjugatformel eine Ausgangsstelle der klauselrepräsentierenden Transition $t_{n(p)}$.

h) Abb. 53 auf der nächsten Seite enthält einige einfache Beispiele für die Topologien $TOP(kla_p)$ von Teilnetzen, die jeweils eine Klausel kla_p abbilden[16)]. Diese Beispiele verdeutlichen, wie durch das gleiche, zuvor beschriebene Konstruktionsschema unterschiedlich strukturierte Klauseln repräsentiert werden können. Die Konstruktionen überdecken das gesamte Spektrum, das mit der Repräsentation eines atomaren Formelvorkommnisses beginnt, über Negate, Kon-, Ad-, Dis-, Bi- sowie Subjugate führt und schließlich bei der Repräsentation allgemeiner Klauseln endet[17)].

i) Die Teilnetze, die jeweils eine Klausel aus dem Objektmodell repräsentieren, werden mittels der Stellenidentifizierung zum Gesamtnetz des Netzmodells zusammengeführt. Dabei gibt die Operation der Stellenidentifizierung die konjunktive Verknüpfung der Klauseln wieder, die das repräsentierte Formelsystem in seiner konjunktiven Normalform besitzt. Die Topologien der beteiligten Teilnetze stellen bereits die aussagenlogische Verknüpfungsstruktur aller involvierten Klauseln dar. Daher kann die Topologie des Gesamtnetzes gedeutet werden als die netzartige Repräsentation des internen aussagenlogischen Zusammenhangs desjenigen Formelsystems, das als Objektmodell den Ausgangspunkt der Netzmodellierung bildete. Der prädikatenlogische Gehalt dieses Formelsystems findet dagegen in den strukturierten Marken des Netzmodells seinen Niederschlag[18)].

j) Aufgrund der Gültigkeits-Prämisse wird das gesamte Wissen, das die Interpretation I_0 über die Wahrheitswerte von Formeln des prädikatenlogischen Objektmodells für dessen Ausgangszustand "r=0" zur Verfügung stellt, durch die Gültigkeit atomarer Formelvorkommnisse ausgedrückt. Diese gültigen Formelvorkommnisse stellen Fakten dar. Falls unter der Interpretation I_0 mindestens ein Fakt bekannt ist, so scheint dies auf den ersten Blick der Markenfreiheit des deklarativen Netzmodells zu widersprechen. Denn bei der Entfaltung Synthetischer Netze wurde dargelegt, daß jedes Fakt $prä_u(gt_1,...,gt_{Ku})$ mit $I_0(prä_u(gt_1,...,gt_{Ku}))=$gültig und $K_u \in \mathcal{N}_+$ als ein K_u-stelliges Markentupel $<gt_1,...,gt_{Ku}>$ auf derjenigen Stelle s_m repräsentiert wird, welche mit dem K_u-stelligen Prädikatssymbol $Prä_u$ für das faktisch gültige atomare Formelvorkommnis $prä_u(gt_1,...,gt_{Ku})$ beschriftet ist. Also müßte der Interpretation I_0, die mindestens ein Fakt umfaßt, ein Netzmodell mit mindestens einem Markentupel entsprechen. Dennoch ist das deklarative Netzmodell per constructionem markenfrei.

k) Der scheinbare Widerspruch aus der voranstehenden Erläuterung läßt sich jedoch auflösen. Denn ein Fakt stellt eine atomare Klausel dar, die aus genau einem atomaren Formelvorkommnis als positivem Literal besteht, dessen Gültigkeit bekannt ist. Für eine solche faktische Klausel gilt also:

$$kla_p \;:\Leftrightarrow\; lit_p$$

$$lit_p \;:\Leftrightarrow\; prä_{u(p)}(gt_1,...,gt_{Ku})$$

$$I_0(prä_{u(p)}(gt_1,...,gt_{Ku})) = \text{gültig}$$

Die faktische Klausel kla_p wird durch ein Teilnetz abgebildet[19)], dessen Topologie $TOP(kla_p)$ nur aus einer klauselrepräsentierenden Transition $t_{n(p)}$[20)], aus einer Ausgangsstelle $s_{m(p)}$ für das atomare Formelvorkommnis $prä_{u(p)}$ und aus einer verknüpfenden Kante besteht:

$$TOP(kla_p) = (\{s_{m(p)}\},\{t_{n(p)}\};\{(t_{n(p)},s_{m(p)})\})$$

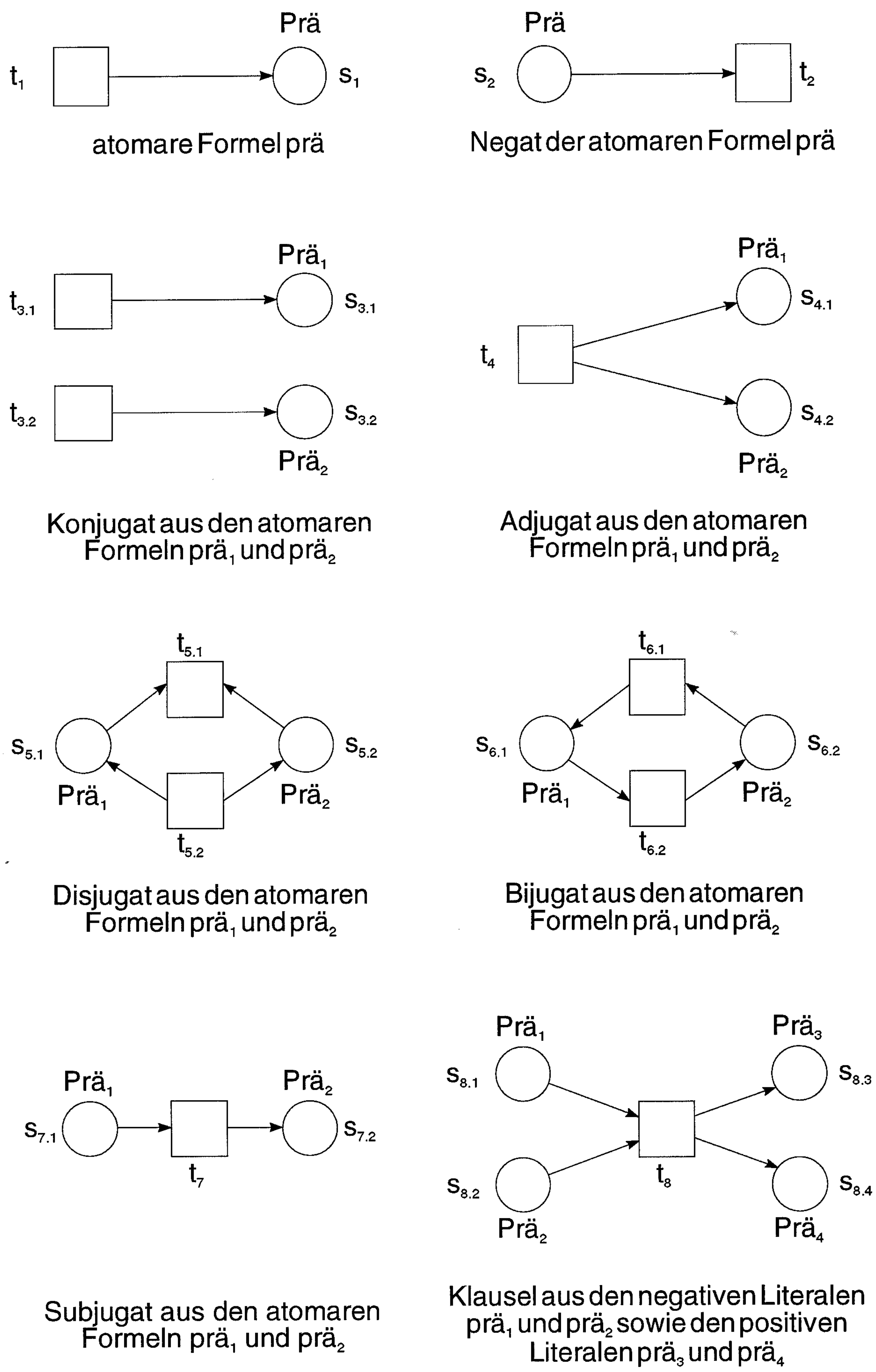

Abb. 53: Netztopologien zur Repräsentation von Klauseln

Die Stelle $s_{m(p)}$ wird mit dem Namen $Prä_u$ desjenigen Prädikatssymbols beschriftet, aus dem das faktisch gültige Formelvorkommnis $prä_u(gt_1,...,gt_{Ku})$ abgeleitet ist. Die faktrepräsentierende Transition $t_{n(p)}$ ist in dem Teilnetz mit der faktspezifischen Topologie $TOP(kla_p)$ per constructionem immer aktiviert[21]. Darüber hinaus zeichnet sich das faktrepräsentierende Teilnetz dadurch aus, daß seine Kante $(t_{n(p)},s_{m(p)})$ mit der Grundtermformel $prä_{u(p)}(gt_1,...,gt_{Ku})$ beschriftet ist. Die Kantenbeschriftungen von Teilnetzen, die keine Fakten darstellen, enthalten dagegen in der Regel auch Variablen.

l) Jedes markenfreie deklarative Netzmodell SN_{FS} für ein Objektmodell in Klauselform $I_0(KL_{FS})$, dessen Interpretation I_0 eine nicht-leere Faktenmenge $FAK_{FS.0}$ festlegt, läßt sich in ein markenbehaftetes äquivalentes Netzmodell SN_{FS}^* transformieren: Zu diesem Zweck wird jede Transition $t_{n(p)}$, die eine faktische Klausel repräsentiert, im deklarativen Netzmodell genau einmal geschaltet. Danach liegt ein Netzmodell mit einer modifizierten Ausgangsmarkierung M_0^* vor. Unter ihr befindet sich auf der Ausgangsstelle $s_{m(p)}$ jeder Transition $t_{n(p)}$, die eine faktische atomare Klausel kla_p repräsentiert, mindestens[22] ein Markentupel. Dann entspricht jedes K_u-stellige Markentupel, das die Stelle $s_{m(p)}$ mit zugeordneten Prädikatssymbol $Prä_u$ belegt, genau einem gültigen atomaren Formelvorkommnis, also genau einem Fakt aus der Interpretation I_0 des prädikatenlogischen Objektmodells. Damit ist der o.a. scheinbare Widerspruch beseitigt.

m) Bei der voranstehenden Einführung einer Netzmarkierung handelt es sich jedoch nur eine Hilfskonstruktion. Sie dient dazu aufzuzeigen, daß sich deklarative Netzmodelle mit der früher vorgelegten Interpretation von Netzmarkierungen in Synthetischen Netzen grundsätzlich vereinbaren lassen. Ansonsten spielt jedoch die modifizierte Ausgangsmarkierung M_0^* für deklarative Netzmodelle keine Rolle. Denn alle prädikatenlogischen Analyseinstrumente, die sich für die Untersuchung von deklarativen Netzmodellen einsetzen lassen, setzen deren Markenfreiheit im Ausgangszustand voraus. Der tiefere Grund für diese Voraussetzung liegt darin, daß die modifizierte Ausgangsmarkierung M_0^* durch jedes Schalten einer Transition untergeht. Die Gültigkeit der Fakten eines deklarativen Netzmodells vermag sich hingegen aufgrund der Monotonie der konventionellen Prädikatenlogik nicht zu verändern: Ein Fakt, das einmal als gültige atomare Grundtermformel bekannt ist, kann in einem deklarativen prädikatenlogischen Objektmodell niemals seine Gültigkeit verlieren[23]. Dies wird durch die oben vorgestellten faktrepräsentierenden Teilnetze mit ihrer invarianten Topologie $TOP(kla_p)$ adäquat modelliert. Die modifizierte Ausgangsmarkierung $M_0^*(s_{m(p)})$ einer Stelle $s_{m(p)}$ aus einem solchen faktrepräsentierenden Teilnetz wird jedoch der prädikatenlogischen Monotonie von deklarativen Netzmodellen nicht gerecht. Denn das Markentupel auf der Stelle $s_{m(p)}$, das unter der modifizierten Ausgangsmarkierung M_0^* noch ein Fakt repräsentiert, kann unter beliebigen Folgemarkierungen durch das Schalten einer Ausgangstransition dieser Stelle abgezogen werden. Die Stelle $s_{m(p)}$ wäre dann nicht mehr durch das Markentupel belegt, obwohl das Fakt weiterhin zur Faktenmenge des deklarativen Objektmodells gehört. Daher werden in deklarativen Netzmodellen faktische Klauseln kla_p im allgemeinen nicht durch die modifizierten, aber *veränderlichen* Ausgangsmarkierungen $M_0^*(s_{m(p)})$ ausgedrückt, sondern durch Teilnetze mit *invarianten* Topologien $TOP(kla_p)$.

n) Eine äquivalente Repräsentation von faktischen Klauseln kla_p beruht auf dem Konzept der Faktnetze[24]. Sie greift nicht auf die invarianten Klauseltopologien $TOP(kla_p)$, sondern auf das spezielle Konstrukt der faktischen Transitionen zurück. Eine faktische Transition ist eine Transition, die unter keiner zulässigen Netzmarkierung aktiviert ist. Eine solche Transition wird auch als tote Transition bezeichnet[25]. Zugleich werden die Gültigkeitsstati atomarer Formelvorkommnisse nicht mehr in der Topologie eines markenfreien deklarativen Netzmodells verborgen. Statt dessen werden sie als Netzmarkierungen explizit ausgewiesen: In einem Faktnetz ist ein atomares Formelvorkommnis genau dann gültig (ungültig), wenn die formelrepräsentierende Stelle markiert (unmarkiert) ist[26].

Aus der oben beschriebenen Konstruktion deklarativer Netzmodelle folgt für jedes klauselrepräsentierende Teilnetz: Die Transition $t_{n(p)}$, die zur Modellierung der Klausel kla_p gehört[27)], ist unter einer Markierung M_r genau dann aktiviert, wenn die Klausel kla_p für alle atomaren Formelvorkommnisse, deren Gültigkeiten die Markierung M_r der inzidenten Stellen $s_{m(p)}$ anzeigt, ungültig ist. Dies resultiert unmittelbar aus folgenden Zusammenhängen[28)]:

- Eine Klausel ist ein Adjugat aus negativen und positiven Literalen.
- Ein Adjugat ist genau dann ungültig, wenn alle seine Komponenten ungültig sind.
- Also ist eine Klausel genau dann ungültig, wenn die atomaren Formelvorkommnisse aus allen seinen negativen Literalen gültig sind und wenn die atomaren Formelvorkommnisse aus allen seinen positiven Literalen ungültig sind.
- Gültige atomare Formelvorkommnisse werden durch entsprechende Markentupel auf denjenigen Stellen abgebildet, welche die zugehörigen Prädikatssymbole darstellen.
- Die Gültigkeit jedes atomaren Formelvorkommnisses aus den negativen Literalen einer Klausel kla_p äußert sich im Netzmodell als ein Markentupel auf der zugehörigen Eingangsstelle der Transition $t_{n(p)}$, welche die Klausel kla_p repräsentiert.
- Die Ungültigkeit jedes atomaren Formelvorkommnisses aus den positiven Literalen einer Klausel äußert sich im Netzmodell als Nichtmarkierung der zugehörigen Ausgangsstelle der Transition $t_{n(p)}$.
- Also sind alle Eingangsstellen einer Transition $t_{n(p)}$ markiert und alle Ausgangsstellen genau dann unmarkiert, falls die Transition die Klausel kla_p repräsentiert und diese Klausel ungültig ist.
- Eine Transition ist genau dann aktiviert, wenn alle ihre Eingangsstellen markiert und ihre Ausgangsstellen unmarkiert sind[29)].
- Folglich ist eine klauselrepräsentierende Transition unter einer Markierung genau dann aktiviert, wenn die Klausel unter derselben Markierung ungültig ist.

Umgekehrt ist die Transition $t_{n(p)}$ aus einem klauselrepräsentierenden Teilnetz niemals aktiviert, falls nur solche Markierungen M_r des deklarativen Netzmodells zugelassen werden, unter denen die Klausel kla_p gültig ist. Diese Einschränkung zulässiger Netzmarkierungen wird dadurch erreicht, daß die Transition $t_{n(p)}$ als eine faktische Transition ausgezeichnet wird[30)]. Dann gilt: Jedes Teilnetz mit genau einer faktischen Transition $t_{n(p)}$ repräsentiert die zugehörige Klausel kla_p so, daß die Klausel unter allen zulässigen Netzmarkierungen gültig ist. Wenn die Klausel als faktische Klausel aus nur genau einem atomaren Formelvorkommnis besteht, so gilt insbesondere auch: Die faktische Klausel ist unter allen zulässigen Netzmarkierungen gültig, weil die zugehörige faktische Transition niemals aktiviert ist. Dies entspricht genau der o.a. prädikatenlogischen Monotonie für deklarative Netzmodelle.

Aufgrund der voranstehenden Erläuterungen lassen sich faktische Klauseln kla_p nicht nur durch invariante Topologien TOP(kla_p) von klauselrepräsentierenden Teilnetzen darstellen. Ebenso ist es möglich, die Transitionen $t_{n(p)}$ derselben Teilnetze als faktische Transitionen auszuzeichnen[31)]. Der Verf. verfolgt hier die letztgenannte Alternative aus zwei Gründen nicht weiter. Erstens setzen die Netztheoreme, die später noch eine Rolle spielen werden, die Faktrepräsentation durch invariante Teilnetztopologien voraus. Denn die Netztheoreme sind für Netze mit faktischen Transitionen überhaupt nicht definiert. Zweitens geht in Faktnetzen der dynamische Charakter von Transitionen, Schaltakte ausführen zu können, durch das Verbot der Transitionsaktivierungen für alle zulässigen Netzmarkierungen weitgehend verloren. Hierin sieht der Verf. eine Einschränkung des Ausdrucksvermögens des Petrinetz-Konzepts, die weder notwendig[32)] noch wünschenswert[33)] ist.

Anmerkungen zum Kapitel:

1) Die Abbildungsvorschrift einer Funktion läßt sich als eine Menge aus 2-Tupeln auffassen: Jede erste Tupelkomponente stellt ein Original aus dem Definitionsbereich der Funktion dar. Jede zweite Tupelkomponente ist das jeweils zugehörige Bild aus dem Wertebereich der Funktion. Die Menge enthält genau so viele 2-Tupel, wie durch die Abbildungsvorschrift der Funktion Paare aus Originalen und zugehörigen Bildern definiert sind.

2) Diese Festlegung könnte auf den ersten Blick erstaunen, weil eine leere Stellenmenge $S=\emptyset$ vorausgesetzt wurde. Dennoch ist sie im formalen Sinne wohldefiniert. Darüber hinaus erweist sie sich auch in dem Sinne als vorteilhaft, als sie später nicht verändert zu werden braucht, wenn die Stellen für die Stellenmenge S eingeführt werden. Denn dann wird jeder Stelle s_m die leere Ausgangsmarkierung $M_0(s_m)=\emptyset$ zugeordnet.

3) Da die Operationenfolgen voneinander unabhängig sind, brauchen ihre Wiederholungen nicht sequentiell aufeinander zu folgen. Statt dessen lassen sich die Operationszyklen auch nebenläufig ausführen.

4) Die Bedingung des zweiten Halbsatzes läßt sich durch zwei technische Maßnahmen sicherstellen: Erstens werden für die Indizes "m" der Stellen s_m, die Indizes "p" der Klauseln kla_p und die Teilindizes "q" der Literale $lit_{p.q}$ jeweils natürliche Zahlen gewählt. Zweitens wird eine Indizierungsfunktion "ind" festgelegt, welche die Klausel- und Literalindizes in *bijektiver* Weise auf Stellenindizes abbildet, so daß jeder Stellenindex m(p.q) genau einem Paar (p.q) aus Klausel- und Literalindex entspricht *und umgekehrt.* Dies läßt sich z.B. erreichen durch folgende Indizierungsfunktion:

$$\text{ind:} \quad \mathcal{N}_+ \times \mathcal{N}_+ \rightarrow \mathcal{N}_+$$

$$(p,q) \rightarrow \text{ind}(p,q) = m(p.q) = (p-1)\bullet Q_{max}+q$$

$$\text{mit:} \quad Q_{max} = \max\{Q_p\text{:}\, p=1,...,P\}$$

Solche Indizierungsfunktionen werden fortan stets unterstellt, auch wenn auf sie nicht mehr ausdrücklich hingewiesen wird. Sie gewährleisten bei der nebenläufigen Ausführung von Operationszyklen, daß bei den Zyklusausführungen nicht zufällig Stellen mit identischen Indizes erzeugt werden. Denn wegen der nebenläufigen Zyklusausführung könnte andernfalls der Fall eintreten, daß zwei Stellen in unterschiedlichen, aber nebenläufig ablaufenden Operationszyklen denselben Index $m(p_1.q_1)=m(p_2.q_2)$ erhalten.

5) Die Operationszyklen ließen sich wiederum nebenläufig ausführen, weil sie für die Klauseln aus der Klauselmenge KL_{PS} unabhängig voneinander erfolgen.

6) Da nur ein Formelvorkommnis betroffen ist, braucht keine Multimenge benutzt zu werden. Eine konventionell definierte Menge würde ausreichen. Dennoch wird hier eine Multimenge verwendet, um die Transformation deklarativer Netzmodelle möglichst kompatibel zur Transformation operationaler Netzmodelle zu gestalten. Bei den letztgenannten läßt sich der Gebrauch von Multimengen nicht vermeiden. Darüber hinaus gestattet die Verwendung von Multimengen, deklarative Objektmodelle so zu transformieren, daß das resultierende Netzmodell unmittelbar die Gestalt eines Synthetischen Netzes annimmt. Andernfalls müßten die Mengen aus der Beschriftung eines deklarativen Netzmodells erst noch in Multimengen eines Synthetischen Netzes transformiert werden, um ein "echtes" Synthetisches Netz zu erhalten.

7) Die Verschiedenheit zweier Formelvorkommnisse $prä_{u(p1.q1)}$ und $prä_{u(p2.q2)}$ wird hier durch die unterschiedliche Subindizierung $(p_1.q_1)\neq(p_2,q_2)$ ausgedrückt. Entweder gehören die beiden Formelvorkommnisse zu unterschiedlichen Klauseln kla_{p1} und kla_{p2} mit $p_1\neq p_2$. Oder sie gehören zwar zur selben Klausel kla_p mit $p=p_1=p_2$, aber zu verschiedenen Literalen $lit_{p.q1}$ und $lit_{p.q2}$ aus dieser Klausel mit $q_1\neq q_2$.

8) Diese Anzahl muß aus der inhaltlichen Analyse des jeweils untersuchten Objektmodells abgeleitet werden. Beispielsweise lassen sich für 1-stellige Formeln $prä_u(\emptyset)$, die als Substitute für 0-stellige Formeln $prä_u()$ eingeführt wurden, keine multiplen Prädikatsgültigkeiten rechtfertigen. Folglich können für diese Formeln $prä_u(\emptyset)$ nur einfache Fakten $fakt_r(1,prä_u(\emptyset))$ definiert werden. Aus der Unmöglichkeit von Fakten $fakt_r(mu_{u.r},prä_u(\emptyset))$ mit Multiplizitäten $mu_{u.r}\geq 2$ folgt für diesen Fall unmittelbar, daß diejenigen Stellen s_m, die den zugrundeliegenden 1-stelligen Prädikatssymbolen $Prä_u$(bas_marke) zugeordnet werden, die Markenkapazitäten $KAP_m=1$ besitzen. Eine Markenkapazität $KAP_m>1$ kann dagegen erforderlich werden, wenn Klauseln zugelassen werden, die mehrere verschiedene Vorkommnisse desselben Prädikatssymbols besitzen. Dann führt die später vorgestellte Stellenidentifizierung dazu, daß die Markenkapazität der Stelle, die mit jenem Prädikatssymbol beschriftet ist, die Markierung durch entsprechend viele Markentupel gestatten muß.
Falls sich keine eindeutige Höchstanzahl gültiger Formelvorkommnisse aus der Modellanalyse ableiten läßt, wird als Voreinstellung die größtmögliche Ganzzahl gewählt, die sich in der Implementierungsumgebung des angestrebten Netzmodells ausdrücken läßt.

9) Vgl. GENRICH (1990b).

10) Die Terme te_k aus dem Argument $(te_1,...,te_{Ku})$ eines Formelvorkommnisses $prä_{u(p.q)}(te_1,...,te_{Ku})$ werden nunmehr mit dem differenzierenden Index "d" versehen und als Terme $te_{k.d}$ notiert, weil sich die Argumente der Formelvorkommnisse $prä_{u(p.q)}(te_1,...,te_{Ku})$ verschiedener Literale $lit_{p.q}$ voneinander unterscheiden müssen. Denn jedes Formelvorkommnis darf in jeder Klausel nur höchstens einmal als Atom *und* höchstens einmal als dessen Negat enthalten sein. Da Literale, die atomare Formelvorkommnisse (positive Literale) oder deren Negate (negative Literale) darstellen, zu jeweils unterschiedlich gerichteten Kanten führen, kann an einem Bündel gleichgerichteter Kante jedes Formelvorkommnis sogar nur höchstens einmal *entweder* als Atom *oder* als dessen Negat vorkommen. Daher muß jedes Formelvorkommnis eines Literals in einem solchen Kantenbündel ein jeweils anderes Argument $(te_1,...,te_{Ku})$ besitzen.

11) Multiplizitäten $mu_{u(p.q)} \geq 2$ können nur bei Produktionsregeln vorkommen. Bei der Transformation der zustandsbeschreibenden Formel $prä_{FS}$ sind sie dagegen ausgeschlossen. Denn in jeder Klausel darf dasselbe atomare Formelvorkommnis $prä_{u(p.q)}$ nur höchstens einmal selbst (positives Literal) oder höchstens einmal als sein Negat (negatives Literal) enthalten sein. Im ersten Fall führt es zu einer Aus-, im zweiten Fall zu einer Eingangskante derjenigen Transition, welche die betrachtete Klausel repräsentiert. Folglich kann jedes atomare Formelvorkommnis an der Spezifizierung jedes Kantengewichts jeweils nur höchstens einmal teilnehmen; q.e.d.

12) Zu einer solchen Multikantenbildung kommt es, wenn in einer Klausel mehrere verschiedene Vorkommnisse desselben Prädikatssymbols so enthalten sind, daß diese Prädikatsvorkommnisse entweder alle zu den positiven oder aber alle zu den negativen Literalen der Klausel gehören. Jedes Prädikatsvorkommnis wurde zunächst durch eine eigene Stelle erfaßt, die mit dem Prädikatssymbol beschriftet ist. Zu jeder dieser Stellen gehört eine Ein- bzw. eine Ausgangskante, deren Gewicht jeweils das Prädikatsvorkommnis mit der Multiplizität $mu_{u,r}=1$ ist. Bei der Stellenidentifizierung entsteht ein Bündel aus gleichgerichteten, aber verschieden beschrifteten Kanten. Diese Kanten werden zu einer Multikante zusammengefaßt, indem die Prädikatsvorkommnisse ihrer Kantengewichte zu einer formalen Summe zusammengefaßt werden. Ein Beispiel für solche Klauseln findet sich bei THIELER-MEVISSEN (1987a), S. 541.

13) Vgl. THIELER-MEVISSEN (1987a), S. 541, Fakt "F6" und seine Abbildung in Fig. 4.

14) In einer früheren Anmerkung wurde bereits für Petrinetze i.e.S. festgelegt, sie mit Hilfe beschrifteter Kanten stets als Monographen darzustellen.

15) Bei einer "klauselrepräsentierenden" Transition handelt es sich jedoch um eine vereinfachte Redeweise, da zur Repräsentation einer Klausel alle Komponenten gehören, die in der oben aufgeführten Transformationsmethode angesprochen wurden. Dazu zählen beispielsweise für jede Klausel kla_p die Netztopologie $TOP(kla_p)$ mit allen ihren Bestandteilen und die Transaktion $tr_{n(p)}$.

16) Analoge Beispiele - allerdings für den einfacheren aussagenlogischen Fall - finden sich bei ZELEWSKI (1989c), S. 23. Vgl. auch die dort ausführlicher diskutierten Netztopologien für die Repräsentation logischer Sachverhalte auf S. 43ff.

17) Die exemplarischen Netztopologien folgen auf nahezu triviale Weise aus der oben dargelegten Transformationsmethode. Lediglich das Disjugat könnte auf den ersten Blick weniger leicht zu durchschauen sein. Es beruht jedoch auf einer einfachen Äquivalenzumformung in zwei konjunktiv verknüpfte Adjugate:

$$prä_1 \veebar prä_2$$

$$\Leftrightarrow \quad (prä_1 \wedge (\neg prä_2)) \vee (\neg prä_1 \wedge prä_2)$$

$$\Leftrightarrow \quad ((prä_1 \wedge (\neg prä_2)) \vee (\neg prä_1)) \wedge ((prä_1 \wedge (\neg prä_2)) \vee prä_2)$$

$$\Leftrightarrow \quad ((prä_1 \vee (\neg prä_1)) \wedge (\neg prä_2 \vee (\neg prä_1)) \wedge (prä_1 \vee prä_2) \wedge (\neg prä_2 \vee prä_2)$$

$$\Leftrightarrow \quad (\neg prä_2 \vee (\neg prä_1)) \wedge (prä_1 \vee prä_2)$$

Eine ausführlich kommentierte Herleitung der Netztopologien findet sich bei ZELEWSKI (1989c), S. 17ff.

18) Der prädikatenlogische Aspekt der Formelquantifizierungen findet keinen expliziten Niederschlag. Denn bei der vorangehenden Erstellung einer konjunktiven Normalform wurden alle Existenzquantoren qua Skolemisierung eliminiert und alle Allquantoren implizit vereinbart. Daher tauchen in der konjunktiven Normalform des Formelsystems, das dem Netzmodell zugrundeliegt, überhaupt keine expliziten Quantoren auf. Entsprechend repräsentiert das Netzmodell die betroffenen Allquantoren auch nur implizit.

Besonders anschaulich wird von THIELER-MEVISSEN (1987a), S. 539ff. u. 552f, beschrieben, auf welche Weise die All- und Existenzquantoren aus prädikatenlogischen Formeln in die formelrepräsentierenden Netze Eingang finden. Dabei zeigt die Autorin auch die Möglichkeit auf, in deklarativen Netzmodellen Existenzquantoren - ohne den Umweg ihrer Skolemisierung - direkt zu erfassen (S. 541ff. u. 552f.). Die Kantengewichte nehmen dann die Gestalt formaler Summen an, die sich über die gesamten Definitionsbereiche der jeweils betroffenen Variablen erstrecken. Diese Konstruktion läßt sich allerdings nur so lange anwenden, wie die Definitionsbereiche endlich bleiben.

19) Vgl. dazu die Repräsentation jeder beliebigen Klausel durch ein Teilnetz mit klauselspezifischer Topologie.

20) Da diese Transition eine faktische Klausel repräsentiert, wird sie auch als faktrepräsentierende Transition angesprochen.

21) Denn für die Stelle $s_{m(p)}$ wurde die Nullmarkierung $M_0(s_{m(p)}):=0$ festgelegt.

22) Es können auch mehrere Markentupel sein. Dies ist genau dann der Fall, wenn mehrere Transitionen faktische Klauseln repräsentieren, deren atomare Formelvorkommnisse jeweils aus demselben Prädikatssymbol $Prä_u$ abgeleitet sind. Dann besitzen diese Transitionen eine gemeinsame Ausgangsstelle, die mit diesem Prädikatssymbol beschriftet sind. Jede dieser Transitionen legt auf der gemeinsamen Ausgangsstelle schaltbedingt genau ein Markentupel ab. Da diese Ausgangsstelle von mehreren Transitionen geteilt wird, müssen dort auch mehrere Markentupel abgelegt werden; q.e.d.

23) Das nachträgliche Ungültigwerden eines vormals gültigen Formelvorkommnisses ist erst dann möglich, wenn auch Veränderungen des Modellzustands zugelassen werden, die sich in Modifizierungen der zustandsspezifischen Faktenmengen niederschlagen. Solche Veränderungen von Faktenmengen werden aber erst in operationalen Objektmodellen erfaßt, in denen zustandsverändernde Aktionen durch Produktionsregeln abgebildet werden.

24) Vgl. zur Konzeption der Faktnetze im allgemeinen PETRI,C. (1976a), S. 75ff.; PETRI,C. (1977d), S. 23ff.
Vgl. speziell zur Repräsentation von prädikaten- und aussagenlogischen Formeln durch Faktnetze THIELER-MEVISSEN (1987a), S. 533ff., insbesondere S. 537ff.; ZELEWSKI (1989c), S. 40ff., 43ff. u. 51ff.; ZELEWSKI (1989e), S. 73f., 82, 85f. u. 88f.; GENRICH (1990b).

25) Vgl. dazu die Definition toter Transitionen im Kontext der Analyse dynamischer Netzeigenschaften.

26) Vgl. zu dieser eineindeutigen Zuordnung zwischen den Gültigkeitsstati eines atomaren Formelvorkommnisses einerseits und den Markierungen einer formelrepräsentierenden Stelle andererseits THIELER-MEVISSEN (1977), S. 8f.; ZELEWSKI (1989c), S. 70; ZELEWSKI (1989e), S. 74.
Hinsichtlich dieser Zuordnung zwischen Gültigkeitsstati und Markierungen stimmen Faktnetze mit den operationalen Netzmodellen überein, die im nächsten Kapitel eingeführt werden.

27) Der Einfachheit halber wird anschließend von der klauselrepräsentierenden Transition gesprochen.

28) Dabei wird vereinfachend unterstellt, daß für jedes Prädikatssymbol in einer Klausel kla_p nur höchstens ein atomares Formelvorkommnis enthalten ist. Dann besitzt die Transition $t_{n(p)}$ aus dem klauselrepräsentierenden Teilnetz keine Multikanten, sondern nur gewöhnliche Ein- und Ausgangskanten. Unter dieser Voraussetzung können die Markenkapazitäten aller inzidenten Stellen $s_{m(p)}$ der Transition $t_{n(p)}$ auf $KAP_{m(p)}=1$ festgelegt werden. Dies läßt unten zu, die Aktivierung der klauselrepräsentierenden Transition in besonders einfacher Weise durch eine notwendige und hinreichende Bedingung auszudrücken. Falls für mindestens ein Prädikatssymbol in derselben Klausel mehrere Formelvorkommnisse enthalten wären, müßten Multikanten und entsprechende Markenkapazitäten $KAP_{m(p)}>1$ berücksichtigt werden. Dies würde die anschließende Argumentation beträchtlich komplizierter ausfallen lassen, ohne zu besonderen Einsichten zu führen. Daher wird auf diese Komplikation verzichtet.

29) Dies gilt nur unter der vereinfachenden Voraussetzung, daß alle adjazenten Kanten der Transition $t_{n(p)}$ keine Multikanten darstellen und alle inzidenten Stellen $s_{m(p)}$ die Markenkapazitäten $KAP_{m(p)}=1$ besitzen.

30) Oben wurde eine Transition, die unter allen zulässigen Markierungen nicht aktiviert ist, als faktische Transition eingeführt.

31) Die Teilnetze, welche die Klauseln kla_p repräsentieren, behalten zwar auch dann ihre invariante Topologie $TOP(kla_p)$. Doch wird auf diese Teilnetztopologie nicht mehr Bezug genommen, um den faktischen Charakter einer Klausel kla_p auszudrücken.

32) Die Einschränkung braucht nicht zu erfolgen, weil alle faktischen Klauseln durch Teilnetze mit invarianten Topologien repräsentiert werden können.

33) Gerade die Möglichkeit des Petrinetz-Konzepts, die Dynamik von Prozessen unmittelbar modellieren zu können, stellt einen seiner charakteristischen Vorzüge dar.

5.1.3.2.3.3 Transformation operationaler Objektmodelle

Methode zur Transformation von operationalen Objektmodellen in Netzmodelle:

a) Die Transformationsmethode[1)] startet mit einem operationalen Objektmodell in der Produktionsregelform $(I_0{}^*(PR_{FS}),FAK_{FS.0})$.

b) Das Definitionstupel $SN_{FS}=(TOP,SPEC_{MSIG};BES,M_0;IB)$ eines Synthetischen Netzes wird für den Ausgangszustand "r=0" des Netzmodells durch folgende Festlegungen initialisiert:

- ❑ Die Stellen-, Transitionen- und Kantenmenge aus der Netztopologie TOP=(S,T;F) sind jeweils leer: $S:=\emptyset$, $T:=\emptyset$ und $F:=\emptyset$.
- ❑ Die Netzspezifikation $SPEC_{MSIG}=(SO,OP,PRÄ;OBF,OPF,FAK_0;TTMF(VAF),VBM;RES;TR;ÜS)$ läßt sich weitgehend aus dem prädikatenlogischen Objektmodell ableiten:
  - Die Sortenmenge SO umfaßt alle Sorten $sort_i$, die im prädikatenlogischen Objektmodell vorkommen. Dabei kann es sich jeweils um Attribute $attribut_q$ für die Repräsentation von Objekteigenschaften oder um sortierte Marken sor_marke_s für die Darstellung von Objektklassen handeln.
  - Die Menge OP enthält alle Operationssymbole Op_j, die im Objektmodell dazu dienen, komplexe Objekteigenschaften oder Objektklassen aus einfacheren Eigenschaften bzw. Klassen zusammenzusetzen.
  - Die Menge PRÄ führt alle Prädikatssymbole $Prä_u$ auf, aus denen im Objektmodell prädikatenlogische Formeln $prä_u$ abgeleitet sind. Die Formelvorkommnisse drücken jeweils zustandsbeschreibende Aussagen, Ausprägungen von Objekteigenschaften oder Objektbeziehungen aus.
  - Die Familie OBF ergibt sich aus den Mengen OB_i formaler Objekte, die im Objektmodell für alle Sorten $sort_i$ festliegen. Die Objektmengen OB_i definieren alle zulässigen Eigenschaftsausprägungen und alle zulässigen Objekte.
  - Die Familie OPF umfaßt alle Operationen op_j, die aus Operationssymbolen Op_j abgeleitet sind. Sie dienen im Objektmodell dazu, die Ausprägungen komplexer Objekteigenschaften oder komplexe Objekte aus einfacheren Eigenschaftsausprägungen bzw. Objekten zusammenzusetzen.
  - Die Familie FAK_0, die für alle Prädikatssymbole $Prä_u$ aus der Menge PRÄ die Faktenmengen $FAK_{u.0}$ umfaßt, kann unmittelbar aus der Produktionsregelform $(I_0(PR_{FS}),FAK_{FS.0})$ des operationalen Objektmodells abgeleitet werden[2)]:

    $$FAK_0 := (FAK_{u.0}: u=1,...,U \wedge FAK_{u.0} \subseteq FAK_{FS.0})$$

    $$\bigcup(u\in\{1,..,U\}): FAK_{u.0} = FAK_{FS.0}$$

  - Die Familie VAF enthält alle sortenspezifischen Variablenmengen VA_i, deren Elemente Variablen aus den Formelvorkommnissen des Objektmodells darstellen.
  - Für die Familie TTMF(VAF) teilevaluierter Terme und für die Menge VBM aller Variablenbindungsfunktionen werden ihre Definitionen aus der Darstellung Synthetischer Netze unverändert übernommen.

- Die Restriktionenmenge RES wird auf die Menge $FOR_{standard}$ aller algebraischen Standardformeln festgesetzt: $RES := FOR_{standard}$.
- Die Transaktionenmenge TR wird als leere Menge initialisiert: $TR := \emptyset$.
- Das allgemeine Übergangsschema ÜS wird wiederum aus der Definition Synthetischer Netze übernommen.

❑ Die Abbildungsvorschriften der Beschriftungsfunktionen bsp, bsk, btt und bfm aus die Netzbeschriftung $BES = \{bsp, bsk, btt, bfm\}$ werden durch binäre Anschriftenmengen BSP, BSK, BTT bzw. BFM dargestellt und jeweils durch die leere Menge eingeführt: $BSP := \emptyset$, $BSK := \emptyset$, $BTT := \emptyset$, $BFM := \emptyset$.

❑ Die Ausgangsmarkierung M_0 bildet die Stellenmenge S auf die Familie FAK_0 der Faktenmengen $FAK_{u.0}$ ab: M_0: $S \rightarrow FAK_0$.

❑ Die Menge IB der Integritätsbedingungen wird aus der Definition Synthetischer Netze unverändert übernommen.

c) Die nachfolgende Operationenfolge für jede Produktionsregel pr_y aus der Produktionsregelmenge PR_{FS} mit $y \in \{1,...,Y\}$, für jede regelzugehörige Voraussetzungsformel $con_{y.v}$ aus der Voraussetzungsmenge CON_y mit $v \in \{1,...,V_y\}$ sowie für jede regelzugehörige Wirkungsformel $act_{y.w}$ aus der Wirkungsmenge ACT_y mit $w \in \{1,...,W_y\}$ wiederholt[3)]:

❑ Falls es sich bei der Voraussetzungsformel $con_{y.v}$ um eine Inklusionsformel $ink_{y.v}$ handelt, wird dafür genau eine formelrepräsentierende Stelle $s_{m(y.v)}$ so eingeführt, daß ihr Index "m(y.v)" die Inklusionsformel $ink_{y.v}$ eindeutig identifiziert. Falls die Voraussetzungsformel eine Bedingungsformel $bed_{y.v}$ darstellt, wird keine Stelle eingeführt und der nachfolgende Schritt übersprungen.

❑ Die Stelle $s_{m(y.v)}$ mit dem Namen $Prä_u$ desjenigen Prädikatssymbols $Prä_u(sor\_marke_{s(u.1)} \ldots sor\_marke_{s(u.Ku)})$ beschriftet, auf das die Inklusionsformel $ink_{y.v}$ durch ihre Faktenmenge $FAK_{u(y.v).r}$ Bezug nimmt:

$$ink_{y.v} :\Leftrightarrow MTAI_{u(y.v).n(y)} \leq FAK_{u(y.v).r}$$
$$\rightarrow \; bsp(s_{m(y.v)}) = Prä_u$$

❑ Falls es sich bei der Wirkungsformel $act_{y.w}$ um eine Modifizierungsformel $mod_{y.w}$ handelt, wird dafür genau eine formelrepräsentierende Stelle $s_{m(y.w)}$ so eingeführt, daß ihr Index "m(y.w)" die Modifizierungsformel $mod_{y.w}$ eindeutig identifiziert. Falls die Wirkungsformel eine Bestimmungsformel $bes_{y.w}$ darstellt, wird keine Stelle eingeführt und der nachfolgende Schritt übersprungen.

❑ Die Stelle $s_{m(y.w)}$ wird mit dem Namen $Prä_u$ desjenigen Prädikatssymbols $Prä_u(sor\_marke_{s(u.1)} \ldots sor\_marke_{s(u.Ku)})$ beschriftet, auf das die Modifizierungsformel $mod_{y.w}$ durch ihre Faktenmenge $FAK_{u(y.w).f}$ Bezug nimmt:

$$((mod_{y.w} :\Leftrightarrow red_{y.w} \wedge red_{y.w} :\Leftrightarrow FAK_{u(y.w).f} := FAK_{u(y.v).r} - MTAV_{u(y.w).n(y)})$$
$$\vee \; (mod_{y.w} :\Leftrightarrow exp_{y.w} \wedge exp_{y.w} :\Leftrightarrow FAK_{u(y.w).f} := FAK_{u(y.v).r} + MTAN_{u(y.w).n(y)}))$$
$$\rightarrow \; bsp(s_{m(y.v)}) = Prä_u$$

d) Die anschließende Operationenfolge wird für jede Produktionsregel pr_y aus der Produktionsregelmenge PR_{FS} mit $y \in \{1,...,Y\}$ ausgeführt[4)]:

❑ Für die Produktionsregel pr_y wird eine regelrepräsentierende Transition $t_{n(y)}$ mit eindeutig identifizierendem Index "n(y)" eingeführt.

- Die Transitionenmenge T wird um die Transition $t_{n(y)}$ erweitert:

  $T := T \cup \{t_{n(y)}\}$

- Die Definition der Umgebung der Transition $t_{n(y)}$ wird durch die leeren Mengen $VB(t_{n(y)}) := \varnothing$ und $NB(t_{n(y)}) := \varnothing$ initialisiert.
- Die Transition $t_{n(y)}$ wird mit dem Transaktionsnamen $tr_{n(y)}$ beschriftet: $btt(t_{n(y)}) = tr_{n(y)}$.
- Die Anschriftenmenge BTT für die Abbildungsvorschrift der Beschriftungsfunktion btt wird um die Transitionsbeschriftung $btt(t_{n(y)}) = tr_{n(y)}$ erweitert:

  $BTT := BTT \cup \{(t_{n(y)}, tr_{n(y)})\}$

- Die Definition der Transaktion $tr_{n(y)}$ wird mit den leeren Mengen $VB(tr_{n(y)}) := \varnothing$, $MTAV_{n(y)} := \varnothing$, $IB(tr_{n(y)}) := \varnothing$, $MTAI_{n(y)} := \varnothing$, $NB(tr_{n(y)}) := \varnothing$, $MTAN_{n(y)} := \varnothing$ und $RES_{n(y)} := \varnothing$ initialisiert.
- Die Transition $t_{n(y)}$, welche die Produktionsregel pr_y repräsentiert, wird mit der Stelle $s_{m(y.v)}$, welche die Inklusionsformel $ink_{y.v}$ darstellt, vermittels einer Informationskante $(s_{m(y.v)}, t_{n(y)})$ verknüpft, falls die Voraussetzungsmenge CON_y der Produktionsregel pr_y die Inklusionsformel $ink_{y.v}$ umfaßt:

  $ink_{y.v} \in CONy \;\rightarrow\; ka_{m(y.v).n(y)} = (s_{m(y.v)}, t_{n(y)})$

- Die Kantenmenge (Flußrelation) F wird um jede Informationskante der Transition $t_{n(y)}$ erweitert:

  $ink_{y.v} \in CONy \;\rightarrow\; F := F \cup \{(s_{m(y.v)}, t_{n(y)})\}$

- Der Vorbereich $VB(t_{n(y)})$ der Transition $t_{n(y)}$ wird um jede Stelle $s_{m(y..v)}$ erweitert, mit der die Transition $t_{n(y)}$ durch eine Informationskante verbunden ist:

  $VB(t_{n(y)}) := VB(t_{n(y)}) \cup \{s_{m(y.v)}: (s_{m(y.v)}, t_{n(y)}) \in F\}$

- Der Informationsbereich $IB(tr_{n(y)})$ der Transaktion $tr_{n(y)}$ wird um jedes Prädikatssymbol $Prä_u$ erweitert, aus dem eine Inklusionsformel $ink_{y.v}$ mit einer Faktenmenge $FAK_{u(y.v).r}$ abgeleitet ist, sofern diese Inklusionsformel in der Voraussetzungsmenge CON_y der Produktionsregel pr_y vorkommt:

  $$IB(tr_{n(y)}) := IB(tr_{n(y)}) \cup \{Prä_u: Prä_u \in PRÄ \wedge ink_{y.v} \in CON_y \wedge \ldots$$
  $$ink_{y.v} :\Leftrightarrow MTAI_{u(y.v).n(y)} \leq FAK_{u(y.v).r}\}$$

- Jede Informationskante $(s_{m(y.v)}, t_{n(y)})$ der Transition $t_{n(y)}$ wird durch die Multimenge $MTAI_{u(y.v).n(y)}$ aus der Definition derjenigen Inklusionsformel $ink_{y.v}$ beschriftet, die in der Voraussetzungsmenge CON_y der Produktionsregel pr_y enthalten ist und für deren Repräsentation die Informationskante eingeführt wurde:

  $$(ink_{y.v} \in CON_y \wedge ink_{y.v} :\Leftrightarrow MTAI_{u(y.v).n(y)} \leq FAK_{u(y.v).r})$$
  $$\rightarrow\; bfm(s_{m(y.v)}, t_{n(y)}) = MTAI_{u(y.v).n(y)}$$

- Die Anschriftenmenge BFM für die Abbildungsvorschrift der Beschriftungsfunktion bfm wird um die Beschriftungen aller Informationskanten der Transition $t_{n(y)}$ erweitert:

$$\begin{aligned} & bfm(s_{m(y.v)},t_{n(y)}) = MTAI_{u(y.v).n(y)} \\ \rightarrow\ & BFM := BFM \cup \{((s_{m(y.v)},t_{n(y)}),MTAI_{u(y.v).n(y)})\} \end{aligned}$$

- Für den gesamten Informationsbereich der Transaktion $tr_{n(y)}$ gilt:

$$\begin{aligned} MTAI_{n(y)} := \{MTAI_{u(y.v).n(y)}\text{: } MTAI_{u(y.v).n(y)} = bfm(s_{m(y.v)},t_{n(y)}) \wedge \ldots \\ bsp(s_{m(y.v)}) = Prä_u \wedge Prä_u \in IB(tr_{n(y)}) \wedge btt(t_{n(y)}) = tr_{n(y)}\} \end{aligned}$$

- Die Restriktionsformelmenge $RES_{n(y)}$ der Transaktion $tr_{n(y)}$ wird um alle nicht-tautologischen Bedingungsformeln $bed_{y.v}$ aus der Voraussetzungsmenge CON_y der Produktionsregel pr_y erweitert:

$$RES_{n(y)} := RES_{n(y)} \cup \{bed_{y.v}\text{: } bed_{y.v} \in CON_y\}$$

- Die Restriktionsformelmenge RES wird um alle nicht-tautologischen Bedingungsformeln $bed_{y.v}$ aus der Voraussetzungsmenge CON_y der Produktionsregel pr_y erweitert, sofern sie nicht schon in der Standardformelmenge $FOR_{standard}$ enthalten sind[5]:

$$\forall(bed_{y.v} \in CON_y)\text{: } bed_{y.v} \neq FOR_{standard} \rightarrow RES := RES\ \{bed_{y.v}\}$$

- Die Transition $t_{n(y)}$, welche die Produktionsregel pr_y repräsentiert, wird mit der Stelle $s_{m(y.w)}$, welche die Modifikationsformel $mod_{y.y}$ darstellt, vermittels einer Eingangskante $(s_{m(y.w)},t_{n(y)})$ verknüpft, falls die Wirkungsmenge ACT_y der Produktionsregel pr_y die Modifikationsformel $mod_{y.y}$ umfaßt und es sich dabei um eine Reduktionsformel $red_{y.w}$ handelt:

$$\begin{aligned} & (mod_{y.w} \in ACTy \wedge mod_{y.w} :\Leftrightarrow red_{y.w}) \\ \rightarrow\ & ka_{m(y.w).n(y)} = (s_{m(y.w)},t_{n(y)}) \end{aligned}$$

- Die Kantenmenge (Flußrelation) F des Netzes wird um jede Eingangskante der Transition $t_{n(y)}$ erweitert:

$$\begin{aligned} & (mod_{y.w} \in ACTy \wedge mod_{y.w} :\Leftrightarrow red_{y.w}) \\ \rightarrow\ & F := F \cup \{(s_{m(y.w)},t_{n(y)})\} \end{aligned}$$

- Der Vorbereich $VB(t_{n(y)})$ der Transition $t_{n(y)}$ wird um jede Stelle $s_{m(y.w)}$ erweitert, mit der die Transition $t_{n(y)}$ durch eine Eingangskante verbunden ist:

$$VB(t_{n(y)}) := VB(t_{n(y)}) \cup \{s_{m(y.w)}\text{: } (s_{m(y.w)},t_{n(y)}) \in F\}$$

- Der Vorbereich $VB(tr_{n(y)})$ der Transaktion $tr_{n(y)}$ wird um jedes Prädikatssymbol $Prä_u$ erweitert, aus dem eine Reduktionsformel $red_{y.w}$ mit einer Faktenmenge $FAK_{u(y.w).f}$ abgeleitet ist, sofern diese Reduktionsformel in der Wirkungsmenge ACT_y der Produktionsregel pr_y vorkommt:

$$\begin{aligned} VB(tr_{n(y)}) := VB(tr_{n(y)}) \cup \{Prä_u\text{: } Prä_u \in PRÄ\ \ red_{y.w} \in ACT_y \wedge \ldots \\ red_{y.w} :\Leftrightarrow FAK_{u(y.w).f} := FAK_{u(y.w).r} - MTAV_{u(y.w).n(y)}\} \end{aligned}$$

- Jede Eingangskante $(s_{m(y.w)}, t_{n(y)})$ der Transition $t_{n(y)}$ wird mit der Multimenge $MTAV_{u(y.w).n(y)}$ aus der Definition derjenigen Reduktionsformel $red_{y.w}$ beschriftet, die in der Wirkungsmenge ACT_y der Produktionsregel pr_y enthalten ist und für deren Repräsentation die Eingangskante eingeführt wurde:

$$(red_{y.w} \in ACT_y \wedge red_{y.w} :\Leftrightarrow FAK_{u(y.w).f} := FAK_{u(y.w).r} - MTAV_{u(y.w).n(y)})$$
$$\rightarrow \quad bfm(s_{m(y.v)}, t_{n(y)}) = MTAV_{u(y.w).n(y)}$$

- Die Anschriftenmenge BFM für die Abbildungsvorschrift der Beschriftungsfunktion bfm wird um die aller Eingangskanten der Transition $t_{n(y)}$ erweitert:

$$bfm(s_{m(p.q)}, t_{n(p)}) = MTAV_{u(p.q).n(p)}$$
$$\rightarrow \quad BFM := BFM \cup \{((s_{m(p.q)}, t_{n(p)}), MTAV_{u(p.q).n(p)})\}$$

- Für den gesamten Vorbereich der Transaktion $tr_{n(y)}$ gilt:

$$MTAV_{n(y)} := \{MTAV_{u(y.w).n(y)}: MTAV_{u(y.w).n(y)} = bfm(s_{m(y.w)}, t_{n(y)}) \wedge \ldots$$
$$bsp(s_{m(y.w)}) = Prä_u \wedge Prä_u \in VB(tr_{n(y)}) \wedge btt(t_{n(y)}) = tr_{n(y)}\}$$

- Die Transition $t_{n(y)}$, welche die Produktionsregel pr_y repräsentiert, wird mit der Stelle $s_{m(y.w)}$, welche die Modifikationsformel $mod_{y.y}$ darstellt, vermittels einer Ausgangskante $(t_{n(y)}, s_{m(y.w)})$ verknüpft, falls die Wirkungsmenge ACT_y der Produktionsregel pr_y die Modifikationsformel $mod_{y.y}$ umfaßt und es sich dabei um eine Expansionsformel $exp_{y.w}$ handelt:

$$(mod_{y.w} \in ACTy \wedge mod_{y.w} :\Leftrightarrow exp_{y.w})$$
$$\rightarrow \quad ka_{n(y).m(y.w)} = (t_{n(y)}, s_{m(y.w)})$$

- Die Kantenmenge (Flußrelation) F des Netzes wird um jede Ausgangskante der Transition $t_{n(y)}$ erweitert:

$$(mod_{y.w} \in ACTy \wedge mod_{y.w} :\Leftrightarrow exp_{y.w})$$
$$\rightarrow \quad F := F \cup \{(t_{n(y)}, s_{m(y.w)})\}$$

- Der Nachbereich $NB(t_{n(y)})$ der Transition $t_{n(y)}$ wird um jede Stelle $s_{m(y.w)}$ erweitert, mit der die Transition $t_{n(y)}$ durch eine Ausgangskante verbunden ist:

$$NB(t_{n(y)}) := NB(t_{n(y)}) \cup \{s_{m(y.w)}: (t_{n(y)}, s_{m(y.w)}) \in F\}$$

- Der Nachbereich $NB(tr_{n(y)})$ der Transaktion $tr_{n(y)}$ wird um jedes Prädikatssymbol $Prä_u$ erweitert, aus dem eine Expansionsformel $exp_{y.w}$ mit einer Faktenmenge $FAK_{u(y.w).f}$ abgeleitet ist, sofern diese Expansionsformel in der Wirkungsmenge ACT_y der Produktionsregel pr_y vorkommt:

$$NB(tr_{n(y)}) := NB(tr_{n(y)}) \cup \{Prä_u: Prä_u \in PRÄ \wedge exp_{y.w} \in ACT_y \wedge \ldots$$
$$exp_{y.w} :\Leftrightarrow FAK_{u(y.w).f} := FAK_{u(y.w).r} + MTAN_{u(y.w).n(y)}\}$$

- Jede Ausgangskante $(t_{n(y)},s_{m(y.w)})$ der Transition $t_{n(y)}$ wird mit der Multimenge $MTAN_{u(y.w).n(y)}$ aus der Definition derjenigen Expansionsformel $exp_{y.w}$ beschriftet, die in der Wirkungsmenge ACT_y der Produktionsregel pr_y enthalten ist und für deren Repräsentation die Ausgangskante eingeführt wurde:

$$(exp_{y.w} \in ACT_y \wedge exp_{y.w} :\Leftrightarrow FAK_{u(y.w).f} := FAK_{u(y.w).r} + MTAN_{u(y.w).n(y)})$$
$$\rightarrow \; bfm(t_{n(y)},s_{m(y.v)}) = MTAN_{u(y.w).n(y)}$$

- Die Anschriftenmenge BFM für die Abbildungsvorschrift der Beschriftungsfunktion bfm wird um die Beschriftungen aller Ausgangskanten der Transition $t_{n(y)}$ erweitert:

$$bfm(t_{n(y)},s_{m(y.w)}) = MTAN_{u(y.w).n(y)}$$
$$\rightarrow \; BFM := BFM \cup \{((t_{n(y)},s_{m(y.w)}),MTAN_{u(y.w).n(y)})\}$$

- Für den gesamten Nachbereich der Transaktion $tr_{n(y)}$ gilt:

$$MTAN_{n(y)} := \{MTAN_{u(y.w).n(y)}: MTAN_{u(y.w).n(y)} = bfm(t_{n(y)},s_{m(y.w)}) \wedge \ldots$$
$$bsp(s_{m(y.w)}) = Prä_u \wedge Prä_u \in NB(tr_{n(y)}) \wedge btt(t_{n(y)}) = tr_{n(y)}\}$$

- Die Restriktionsformelmenge $RES_{n(y)}$ der Transaktion $tr_{n(y)}$ wird um alle Bestimmungsformeln $bes_{y.w}$ aus der Wirkungsmenge ACT_y der Produktionsregel pr_y erweitert:

$$RES_{n(y)} := RES_{n(y)} \cup \{bes_{y.w}: bes_{y.w} \in ACT_y\}$$

- Die Restriktionsformelmenge RES wird um alle Bestimmungsformeln $bes_{y.w}$ aus der Wirkungsmenge ACT_y der Produktionsregel pr_y erweitert, sofern sie nicht schon in der Standardformelmenge $FOR_{standard}$ enthalten sind:

$$\forall(bes_{y.w} \in ACT_y): bes_{y.w} \neq FOR_{standard} \;\rightarrow\; RES := RES \cup \{bes_{y.w}\}$$

- Die Transitionenmenge TR des Synthetischen Netzes wird um die Transaktion $tr_{n(y)}$ erweitert:

$$TR := TR \cup \{tr_{n(y)}\}$$

e) Alle formelrepräsentierenden Stellen $s_{m(y.q)}$, die mit $q \in \{v,w\}$ zwar für verschiedene Inklusionsformeln $ink_{y.v}$ oder Modifikationsformeln $mod_{y.w}$ eingeführt wurden, aber jeweils mit demselben Prädikatssymbol $Prä_u$ beschriftet sind, werden miteinander identifiziert:

$$bsp(s_{m(p1.q1)}) = bsp(s_{m(p2.q2)}) = Prä_u \;\rightarrow\; s_{m(p1.q1)} = s_{m(p2.q2)}$$

f) Die nachstehende Operationenfolge wird für jedes Fakt ausgeführt, das in der Faktenmenge $FAK_{FS.0}$ entweder als faktische Formel $fakt_0(prä_u(gt_1,\ldots,gt_{Ku}))$ explizit notiert wird oder als atomare Grundtermformel $prä_u(gt_1,\ldots,gt_{Ku})$ enthalten ist, deren Gültigkeit implizit unterstellt wird:

- Für die faktische Formel $fakt_0(prä_u(gt_1,\ldots,gt_{Ku}))$ oder die gültige atomare Grundtermformel $prä_u(gt_1,\ldots,gt_{Ku})$ wird eine formelrepräsentierende[6)] Stelle $s_{m(u)}$ so eingeführt, daß ihr Index "m(u)" die Formel $prä_u$ oder den Formelbestandteil $prä_u$ eindeutig identifiziert.

- Die Stelle $s_{m(u)}$ wird mit dem Namen $Prä_u$ desjenigen Prädikatssymbols $Prä_u(sor\_marke_{s(u.1)} \ldots sor\_marke_{s(u.Ku)})$ beschriftet, aus dem die faktische Formel oder die gültige atomare Grundtermformel abgeleitet worden ist:

$$(fakt_0(prä_u(gt_1,\ldots,gt_{Ku}))\in FAK_{FS.0} \wedge prä_u(gt_1,\ldots,gt_{Ku})\in FAK_{FS.0})$$
$$\rightarrow \quad bsp(s_{m(u)}) = Prä_u$$

- Für das Fakt $fakt_0(prä_u(gt_1,\ldots,gt_{Ku}))$ oder $prä_u(gt_1,\ldots,gt_{Ku})$ wird eine faktrepräsentierende Transition $t_{n(u)}$ mit eindeutig identifizierendem Index "n(u)" eingeführt.
- Die Transitionenmenge T des Synthetischen Netzes wird um die Transition $t_{n(u)}$ erweitert:

$$T := T \cup \{t_{n(u)}\}$$

- Die Umgebung der Transition $t_{n(u)}$ wird durch die leeren Mengen $VB(t_{n(u)}):=\emptyset$ und $NB(t_{n(u)}):=\emptyset$ initialisiert.
- Die Transition $t_{n(u)}$ wird mit dem Transaktionsnamen $tr_{n(u)}$ beschriftet: $btt(t_{n(u)})=tr_{n(u)}$.
- Die Anschriftenmenge BTT für die Abbildungsvorschrift der Beschriftungsfunktion btt wird um die Beschriftung der Transition $t_{n(u)}$ mit der Transaktion $tr_{n(u)}$ erweitert:

$$BTT := BTT \cup \{(t_{n(u)},tr_{n(u)})\}$$

- Die Definition der Transaktion $tr_{n(u)}$ wird mit den leeren Mengen $VB(tr_{n(u)}):=\emptyset$, $MTAV_{n(u)}:=\emptyset$, $IB(tr_{n(u)}):=\emptyset$, $MTAI_{n(u)}:=\emptyset$, $NB(tr_{n(u)}):=\emptyset$, $MTAN_{n(u)}:=\emptyset$ und $RES_{n(u)}:=\emptyset$ initialisiert.
- Die Transition $t_{n(u)}$, die das Fakt $fakt_0(prä_u(gt_1,\ldots,gt_{Ku}))$ oder $prä_u(gt_1,\ldots,gt_{Ku})$ repräsentiert, wird mit der Stelle $s_{m(u)}$, welche die Formel $prä_u$ darstellt, vermittels einer Ausgangskante $(t_{n(u)},s_{m(u)})$ verknüpft:

$$(fakt_0(prä_u(gt_1,\ldots,gt_{Ku}))\in FAK_{FS.0} \wedge prä_u(gt_1,\ldots,gt_{Ku})\in FAK_{FS.0})$$
$$\rightarrow \quad ka_{n(u).m(u)} = (t_{n(u)},s_{m(u)})$$

- Die Kantenmenge (Flußrelation) F des Netzes wird um die Ausgangskante der Transition $t_{n(u)}$ erweitert:

$$(fakt_0(prä_u(gt_1,\ldots,gt_{Ku}))\in FAK_{FS.0} \wedge prä_u(gt_1,\ldots,gt_{Ku})\in FAK_{FS.0})$$
$$\rightarrow \quad F := F \cup \{(t_{n(u)},s_{m(u)})\}$$

- Der Nachbereich $NB(t_{n(u)})$ der Transition $t_{n(u)}$ wird um die Stelle $s_{m(u)}$ erweitert, mit der die Transition $t_{n(u)}$ durch die Ausgangskante $(t_{n(u)},s_{m(u)})$ verbunden ist:

$$NB(t_{n(u)}) := NB(t_{n(u)}) \cup \{s_{m(u)}: (t_{n(u)},s_{m(u)})\in F\}$$

- Der Nachbereich $NB(tr_{n(u)})$ der Transaktion $tr_{n(u)}$ wird um das Prädikatssymbol $Prä_u$ erweitert, aus dem das Fakt $fakt_0(prä_u(gt_1,\ldots,gt_{Ku}))$ oder $prä_u(gt_1,\ldots,gt_{Ku})$ abgeleitet ist:

$$NB(tr_{n(u)}) := NB(tr_{n(u)}) \cup \{Prä_u\!: Prä_u\in PRÄ \wedge \ldots$$
$$(fakt_0(prä_u(gt_1,\ldots,gt_{Ku}))\in FAK_{FS.0} \wedge prä_u(gt_1,\ldots,gt_{Ku})\in FAK_{FS.0})\}$$

- Die Ausgangskante $(t_{n(u)},s_{m(u)})$ der Transition $t_{n(u)}$ wird durch die Multimenge $MTAN_{u.n(u)}$ beschriftet, welche die atomare Grundtermformel aus dem Fakt $fakt_0(prä_u(gt_1,...,gt_{Ku}))$ oder $prä_u(gt_1,...,gt_{Ku})$ genau einmal umfaßt:

$$\begin{aligned} &((fakt_0(prä_u(gt_1,...,gt_{Ku}))\in FAK_{FS.0} \wedge prä_u(gt_1,...,gt_{Ku})\in FAK_{FS.0}) \\ &\wedge MTAN_{u.n(u)} = \{1,prä_u(gt_1,...,gt_{Ku})\}) \\ \rightarrow\ &bfm(t_{n(u)},s_{m(u)}) = MTAN_{u.n(u)} \end{aligned}$$

- Die Anschriftenmenge BFM für die Abbildungsvorschrift der Beschriftungsfunktion bfm wird um die Beschriftung der Ausgangskante der Transition $t_{n(u)}$ erweitert:

$$\begin{aligned} &bfm(t_{n(u)},s_{m(u)}) = MTAN_{u.n(u)} \\ \rightarrow\ &BFM := BFM \cup \{((t_{n(u)},s_{m(u)}),MTAN_{u.n(u)})\} \end{aligned}$$

- Für den gesamten Nachbereich der Transaktion $tr_{n(u)}$ gilt: $MTAN_{n(u)} := MTAN_{u.n(u)}$.

g) Alle Stellen $s_{m(p.q)}$ und $s_{m(u)}$, die $bsp(s_{m(p.q)}) = Prä_u$ bzw. $bsp(s_{m(u)}) = Prä_u$ mit dem Namen $Prä_u$ desselben Prädikatssymbols beschriftet sind, werden miteinander identifiziert und durch genau eine Stelle ersetzt. Danach liegt für jedes Prädikatssymbol genau eine Stelle vor, die mit dem Namen des Prädikatssymbols beschriftet ist. Die Indizierung aller Stellen wird derart erneuert, daß die neuen Indizes "m" die Werte $m = 1,...,M$ lückenlos durchlaufen. Die Stellenmenge S des Netzes ist dann die Menge aller reindizierten Stellen s_m.

h) Die ursprünglich leere Anschriftenmenge BSP für die Abbildungsvorschrift der Beschriftungsfunktion bsp wird um die Beschriftungen aller Stellen s_m aus der Stellenmenge S mit Prädikatssymbolen $Prä_u$ erweitert, die nach der Stellenidentifizierung und Reindexierung vorliegen:

$$bsp(s_m) = Prä_u \rightarrow BSP := BSP \cup \{(s_m,Prä_u)\}$$

i) Für jede Stelle s_m aus der Stellenmenge S, der mittels der Beschriftungsfunktion $bsp(s_m) = Prä_u$ das Prädikatssymbol $Prä_u$ zugeordnet ist, wird festgelegt, wie viele gültige Formelvorkommnisse $prä_u$ unter jeder beliebigen Interpretation I_r des prädikatenlogischen Objektmodells maximal existieren dürfen. Diese obere Schranke der Kardinalität der prädikatszugehörigen Faktenmenge $FAK_{u.r}$ ist die Markenkapazität KAP_m der Stelle s_m mit $KAP_m \in \mathcal{N}_+$. Die Stelle s_m wird mit ihrer Markenkapazität beschriftet: $bsk(s_m) = KAP_m$.

j) Die ursprünglich leere Anschriftenmenge BSK für die Abbildungsvorschrift der Beschriftungsfunktion bsk wird um die Beschriftungen aller Stellen s_m aus der Stellenmenge S mit ihren Markenkapazitäten KAP_m erweitert:

$$bsk(s_m) = KAP_m \rightarrow BSK := BSK \cup \{(s_m,KAP_m)\}$$

k) Jeder Stelle $s_m \in S$ wird die Faktenmenge $FAK_{u.0}$ zugeordnet, die bereits bei der Initialisierung der Netzspezifikation aus der Faktenmenge $FAK_{FS.0}$ des Objektmodells in Produktionsregelform gewonnen worden ist: $M_0(s_m) := FAK_{u.0}$. Im Regelfall wird mindestens eine dieser Stellenmarkierungen $M_0(s_m)$ eine nicht-leere Faktenmenge $FAK_{u.0} \neq \emptyset$ sein. Dann liegt ein markenbehaftetes Netz vor. Nur für den Ausnahmefall, daß zufällig alle Faktenmengen $FAK_{u.0}$ leer sind, resultiert ein markenfreies Netz mit $M_0(s_m) = \emptyset$ für alle Stellen $s_m \in S$. Solange unbekannt ist, ob tatsächlich der Regelfall eines markenbehafteten oder der Ausnahmefall eines markenfreien Netzes zutrifft, wird von einem potentiell markenbehafteten Netz gesprochen.

l) Eine Zusammenfassung von Kantenbündeln zu jeweils einer Multikante, wie sie bei der Transformation von deklarativen Objektmodellen erforderlich werden konnte, entfällt für operationale Objektmodelle. Denn die Produktionsregeln wurden mit Multimengen für ihre Inklusions-, Reduktions- und Expansionsformeln von vornherein so definiert, daß zu jedem Prädikatssymbol $Prä_u$ höchstens eine Inklusions-, höchstens eine Reduktions- und höchstens eine Expansionsformel existiert. Zu Inklusions-, Reduktions- und Expansionsformeln gehören verschiedenartige Informations-, Eingangs- bzw. Ausgangskanten. Daher ist es ausgeschlossen, daß zwischen derselben Stelle und derselben Transition mehrere gleichgerichtete Kanten derselben Kantenart ein Kantenbündel bilden.

m) Das Synthetische Netz $SN_{FS} = (TOP,SPEC_{MSIG};BES,M_0;IB)$, das nach allen voranstehenden Transformationsschritten vorliegt, ist das gesuchte operationale Netzmodell für das Objektmodell in Produktionsregelform $(I_0{}^*(PR_{FS}),FAK_{FS.0})$.

Erläuterungen und Ergänzungen zur Transformationsmethode:

a) Bei jedem operationalen Netzmodell handelt sich um ein potentiell markenbehaftetes Netzmodell.

b) Jede Transition repräsentiert entweder genau eine Produktionsregel oder genau ein Fakt aus dem zugrundeliegenden Objektmodell. Wegen dieser bijektiven Zuordnungsmöglichkeit zwischen Transitionen und Produktionsregeln wird auch von regelspezifischen, -zugehörigen oder -repräsentierenden[7)] Transitionen gesprochen.

c) Jede Stelle gibt genau ein Prädikatssymbol aus dem Objektmodell wieder. Sie wird daher auch als prädikatspezifische, -zugehörige oder -repräsentierende Stelle bezeichnet.

d) Für die Inklusionsformeln aus der Regelvoraussetzung einer Produktionsregel bestehen in einem operationalen Netzmodell zwei alternative Repräsentationsmöglichkeiten. Beiden gemeinsam ist, daß die Inklusionsformeln den Charakter von Nebenbedingungen besitzen: Sie müssen gültig sein, damit die Produktionsregel ausgeführt werden kann. Aber sie verlieren durch die Regelwirkung ihre Gültigkeit nicht. Die erste, konventionelle Möglichkeit stellt die Inklusionsformeln als 1-Schleifen dar[8)]. Dabei wird die Stelle, die mit einer Inklusionsformel aus der Regelvoraussetzung beschriftet ist, mit der regelrepräsentierenden Transition sowohl durch eine Ein- als auch durch eine Ausgangskante verknüpft. Dadurch läßt das Schalten der regelrepräsentierenden Transition die Markierung derjenigen Stelle, auf welche die Inklusionsformel abgebildet wurde, unverändert[9)]. Entsprechend wird die Gültigkeit der Regelvoraussetzung von Produktionsregeln aufrechterhalten, wenn diese Regeln ausgeführt werden. Solche 1-Schleifen für jede Inklusionsformel aus den Regelvoraussetzungen von Produktionsregeln sind jedoch aufwendig. Sie vermindern auch die Netztransparenz. Darüber hinaus leiden sie unter dem Mangel, daß sie ein konzeptionelles Repräsentationsdefizit von Nebenbedingungen nicht grundsätzlich beseitigen, sondern lediglich kompensatorisch verdecken[10)].

Daher wählt der Verf. bei der Konstruktion von Netzmodellen die zweite Möglichkeit für die Repräsentation von Inklusionsformeln aus Produktionsregeln. Sie beruht auf der unkonventionellen Verwendung von Informationsstellen und -kanten: Jede Inklusionsformel aus der Regelvoraussetzung einer Produktionsregel wird im Netzmodell wiederum durch eine Stelle repräsentiert, die mit dem Prädikatssymbol der Inklusionsformel beschriftet ist. Diese Stelle gehört jedoch zum Informationsbereich der regelrepräsentierenden Transition. Über die Informationskante zwischen dieser Informationsstelle und der Transition wird keine Marke abgezogen, wenn die regelrepräsentierende Transition im Netzmodell geschaltet wird. Folglich wird die Gültigkeit der Inklusionsformel, die auf die betrachtete Informationsstelle abgebildet wurde, durch die Regelausführung grundsätzlich nicht betroffen. Eine nachträgliche Kompensation von Repräsentationsmängeln, wie sie zuvor bei der Verwendung von 1-Schleifen erforderlich war, erübrigt sich daher. Darüber hinaus fällt das Konstrukt aus Informationsstelle und -kante kompakter als eine analoge 1-Schleife aus. Aufgrund dieser Vorteile hat sich der Verf. für diese zweite Alternative entschieden, Produktionsregeln mit der Hilfe von Informationsstellen und -kanten zu modellieren.

e) Für operationale Netzmodelle und ihre zugrundeliegenden prädikatenlogischen Objektmodelle besteht ein unmittelbarer und anschaulicher Zusammenhang zwischen Netzmarkierungen und Modellinterpretationen: Jede Netzmarkierung M_r korrespondiert mit genau einer Interpretation I_r des Objektmodells vice versa. Dabei wird jedes Prädikatssymbol $Prä_u(sor\_marke_{s(u.1)}, ..., sor\_marke_{s(u.Ku)})$ aus dem Objektmodell einer Stelle s_m aus dem Netzmodell eineindeutig zugeordnet, indem die Stelle mit dem Prädikatssymbolnamen $Prä_u$ beschriftet wird: $bsp(s_m) = Prä_u$. Der Einfachheit halber wird hier zunächst nur ein einstelliges Prädikatssymbol $Prä_u(sor\_marke_{s(u.1)})$ betrachtet, dessen Argument für eine Attributmarke $att\_marke_{s(j)}$ mit $att\_marke_{s(j)} = sor\_marke_{s(u.1)}$ definiert ist. Jede Kopie $m_{s(j).d} = marke_j(at_{j.1.d}, ..., at_{j.Kj.d})$ der Attributmarke, die sich auf dieser Stelle unter einer Netzmarkierung M_r befindet, drückt aus, daß das Prädikatsvorkommnis $prä_u(m_{s(j).d}) = prä_u(marke_j(at_{j.1.d}, ..., at_{j.Kj.d}))$ unter der Interpretation I_r als gültig bekannt ist. Das Fakt $fakt_r(mu_{u.r}, prä_u(marke_j(at_{j.1.d}, ..., at_{j.Kj.d})))$ gibt durch seine Multiplizität $mu_{u.r}$ an, wie viele identische Markenkopien $m_{s(j).d}$ sich auf der Stelle s_m unter der Markierung M_r befinden. Zugleich handelt es sich um die gültigen identischen Vorkommnisse $prä_u(m_{s(j).d})$ des Prädikatssymbols $Prä_u$, die unter der Interpretation I_r in der Faktenmenge $FAK_{u.r}$ dieses Prädikatssymbol enthalten sind.

Daraus folgt eine eineindeutige Entsprechung zwischen einerseits der Faktenmenge $FAK_{u.r}$, die in einem Objektmodell für ein Prädikatssymbol $Prä_u$ unter einer Interpretation I_r definiert ist, und andererseits der Stellenmarkierung $M_r(s_m)$, die in einem Netzmodell für die prädikatspezifische Stelle s_m unter der interpretationsabbildenden Markierung M_r gilt: Jede Markenkopie $m_{s(j).d}$ aus der Stellenmarkierung $M_r(s_m)$ bedeutet genau ein Argument, welches das Prädikatssymbol $Prä_u$ unter der Interpretation I_r erfüllt (vice versa). Jede Markenkopie $m_{s(j).d}$ auf der Stelle s_m unter der Markierung M_r entspricht also genau einem Prädikatsvorkommnis $prä_u(m_{s(j).d})$, dessen Gültigkeit unter der Interpretation I_r bekannt ist (und umgekehrt). Die Wahrheitswerte aller anderen Prädikatsvorkommnisse $prä_u(ob)$ sind dagegen unter der Interpretation I_r unbekannt, wenn die formalen Objekte "ob" aus ihren Argumenten keine Markenkopien auf der prädikatszugehörigen Stelle s_m unter der aktuellen Markierung M_r darstellen.

Wenn das zuvor betrachtete einstellige Prädikatssymbol mit dem Namen $Prä_u$ nicht für eine Attributmarke $att\_marke_{s(j)}$, sondern für eine Kompositmarke $str\_marke_{s(j)}$ definiert ist, gelten die voranstehenden Ausführungen für das Prädikatssymbol $Prä_u(str\_marke_{s(j)})$ im Prinzip unverändert. Falls dagegen mehrstellige Prädikatssymbole $Prä_u(sor\_marke_{s(u.1)}, ..., sor\_marke_{s(u.Ku)})$ für beliebige sortierte Marken $sor\_marke_{s(u.k)}$ mit $k \in \{1, ..., K_u\}$ und $K_u \geq 2$ vorausgesetzt werden, müssen lediglich alle Bezugnahmen auf einzelne Markenkopien $m_{s(j)}$ durch entsprechende Referenzen auf K_u-Tupel $(m_{s(u.1)}, ..., m_{s(u.Ku)})$ aus mehreren Markenkopien ersetzt werden. Der Einfachheit halber werden sowohl die mehrstelligen K_u-Tupel aus Markenkopien als auch die einzelnen Markenkopien gemeinsam als Markentupel angesprochen[11)].

f) Es wurde voranstehend aufgezeigt, daß in einem operationalen Netzmodell die Markentupel, die eine Stelle mit zugeordnetem Prädikatssymbol unter der Netzmarkierung M_r belegen, jeweils mit Fakten für dasselbe Prädikatssymbol unter einer Interpretation I_r des zugrundeliegenden prädikatenlogischen Objektmodells in eineindeutiger Weise korrespondieren[12)]. Diese wechselseitige Entsprechung zwischen Markentupeln und Fakten stimmt genau mit der Konstruktion der markenbezogenen algebraisch-prädikatenlogischen Spezifikation eines Synthetischen Netzes überein. Daher bieten operationale Netzmodelle den Vorzug, daß jede ihrer erreichbaren Markierungen M_r genau einer wohldefinierten Interpretation I_r des zugrundeliegenden operationalen Objektmodells entspricht (und umgekehrt). Wenn der visualisierte Graph des Netzes benutzt wird, erfolgt die Wiedergabe der Modellinterpretation I_r durch die Netzmarkierung M_r in besonders anschaulicher Weise.

Anmerkungen zum Kapitel:

1) Alle Vereinbarungen, die für die Transformation deklarativer Objektmodelle in Klauselform eingeführt wurden, werden nachfolgend als bekannt unterstellt. Sie werden in dem Ausmaß übernommen, in dem die Transformation operationaler Objektmodelle keine neuartigen, an ihre Produktionsregelform angepaßten Vereinbarungen erfordert.

2) Die Ableitung betrifft lediglich den formalen Aspekt, die Faktenmenge $FAK_{FS.0}$ aus dem operationalen Objektmodell in eine Familie FAK_0 aus prädikatssymbolspezifischen Faktenmengen $FAK_{u.0}$ für das Netzmodell zu transformieren. Die erste Transformationsoperation leistet die Separation der Faktenmenge $FAK_{FS.0}$ in Teilmengen $FAK_{u.0}$ für jedes Prädikatssymbol $Prä_u$ aus der Menge PRÄ. Die zweite Transformationsoperation stellt sicher, daß jeweils maximale Teilmengen gebildet werden, die für jedes Prädikatssymbol alle Fakten enthalten. Die involvierten Faktenmengen stellen jeweils Multimengen dar.

3) Die voneinander unabhängigen Operationenfolgen lassen sich wiederum als nebenläufige Operationszyklen ausführen.

4) Die Operationszyklen ließen sich wiederum nebenläufig ausführen, weil sie für die Produktionsregeln aus der Regelmenge PR_{FS} unabhängig voneinander erfolgen.

5) Die tautologische Formel ist in der Standardformelmenge immer enthalten.

6) Bei der formelrepräsentierenden Stelle handelt es sich zugleich um eine faktrepräsentierende Stelle, da das Fakt nur aus genau einem atomaren Formelvorkommnis besteht. Um den Anschluß an die Terminologie für die Repräsentation von Produktionsregeln zu wahren, wird jedoch nur von einer formelrepräsentierenden Stelle gesprochen.

7) Bei einer "regelrepräsentierenden" Transition handelt es sich um eine vereinfachte Redeweise, da zur Repräsentation einer Produktionsregel alle Komponenten gehören, die in der oben aufgeführten Transformationsmethode angesprochen wurden. Dazu zählen beispielsweise das allgemeine Übergangsschema und für jede Produktionsregel pr_y die Transaktion $tr_{n(y)}$.

8) Vgl. zu dieser Verfahrensweise z.B. BAUMAN (1986), S. 193f.

9) Vgl. dazu die Beschreibung von 1-Schleifen für Nebenbedingungen.

10) Diese Nachteile wurden bereits an früherer Stelle erörtert.

11) Einzelne Markenkopien $m_{s(j)}$ stellen dabei den Grenzfall eines einstelligen Markentupels $(m_{s(u.Ku)})$ mit $K_u = 1$ und $m_{s(u.1)} = m_{s(j)}$ dar.

12) Diese bijektive Zuordnung zwischen Stellenmarkierungen und prädikatspezifischen Faktenmengen gilt a fortiori auch für die Ausgangsmarkierung M_0 eines operationalen Netzmodells und die Interpretation I_0 des deklarativen Objektmodells, aus dem das Netzmodell hervorgegangen ist.

5.1.3.2.3.4 Schwierigkeiten bei der Objektmodellierung

Die Umwandlung der natürlichsprachlichen Konzeptualisierung eines modellierungsrelevanten Realitätsausschnitts in ein Netzmodell ist zweifach unterbestimmt. Dieser Eindeutigkeitsmangel wird aber nicht durch die Transformation von Objektmodellen in Netzmodelle verursacht. Vielmehr liegt er bereits in Freiheitsgraden beim prädikatenlogischen Entwurf der Objektmodelle begründet. Daher stellen die nachfolgend erläuterten Unterbestimmtheiten kein Spezifikum Synthetischer Netze dar. Statt dessen gelten sie für alle prädikatenlogisch fundierten Modellierungskonzepte. Dennoch bilden sie einen beachtenswerten Aspekt für die spätere Beurteilung der Eignung des Petrinetz-Konzepts für betriebswirtschaftliche Modellierungsaufgaben. Denn das Petrinetz-Konzept vermag sein volles Ausdruckspotential erst in Verbindung mit einer prädikatenlogisch basierten Objektmodellierung zu entfalten.

Die erste Unterbestimmtheit der Objektmodellierung resultiert aus dem Umstand, daß seitens der Prädikatenlogik für Eigenschaften (Attribute) realer Objekte zwei formal verschiedene, aber materiell gleichwertige Formulierungsoptionen angeboten werden[1]. Sie wird daher als attributive Unterbestimmtheit bezeichnet. Sie gründet sich auf folgenden Freiheitsgrad:

- Entweder werden Objekteigenschaften mit der Hilfe von zusammengesetzten Termen repräsentiert. Die Prädikatssymbole besitzen dann notwendig tief strukturierte Argumentdefinitionen. Ein reales Objekt stellt einen zusammengesetzten Term[2] $te_{K+1}=fun(te_1,...,te_K)$ dar, dessen Subterme te_k mit $k \in \{1,...,K\}$ jeweils eine Objekteigenschaft ausdrücken. Auf dieser Eigenschaftsmodellierung beruhen z.B. der objektorientierte Gestaltungsansatz und die Verwendung von Attributmarken.
- Oder es werden 1-stellige Prädikatssymbole mit flach definierten Argumenten verwendet. In diesem Fall wird ein reales Objekt durch einen atomaren Term "te" repräsentiert. Ein 1-stelliges Prädikatssymbol $Eig(sort_{i(1)})$ drückt als Formel eig(te) aus, daß dem Objekt "te" aus der Sorte $sort_{i(1)}$ die Eigenschaft "Eig" zukommt. Diese Art, Eigenschaften von Objekten zu formulieren, herrscht in konventionellen prädikatenlogischen Erörterungen vor[3].

Welche dieser beiden Formulierungsoptionen gewählt wird, steht zur Disposition des Modellierungsträgers. Das Dispositionsergebnis, das sich in der jeweils ausgewählten prädikatenlogischen Objektmodellierung niederschlägt, trägt letztlich willkürlichen Charakter.

Wie der Freiheitsgrad bei der Ausfüllung der attributiven Unterbestimmtheit ausgenutzt wird, ist theoretisch unerheblich. Dies gilt allerdings nur in der Hinsicht, daß sich beide Formulierungsweisen prädikatenlogischer Objektmodelle mit der o.a. Transformationsmethode in äquivalente[4] Netzmodelle überführen lassen. Trotzdem besitzen die resultierenden Netzmodelle einen durchaus beachtenswerten praktischen Unterschied. Denn der Gebrauch zusammengesetzter Terme führt einerseits zu relativ[5] komplexen Definitionen zulässiger formaler Objekte in der algebraisch-prädikatenlogischen Spezifikation $SPEC_{MSIG}$ eines Netzmodells. Andererseits fällt die Netztopologie TOP vergleichsweise kompakt aus. Denn es werden keine Stellen mit zugeordneten 1-stelligen Prädikatssymbolen erforderlich, um Objekteigenschaften zu repräsentieren. Umgekehrt bedeutet die Verwendung von 1-stelligen Prädikatssymbolen, daß sich die zulässigen formalen Objekte in der Netzspezifikation $SPEC_{MSIG}$ relativ kompakt definieren lassen. Die Netztopologie TOP wird hingegen durch die eigenschaftsdarstellenden Prädikatssymbole aufgebläht. Folglich besteht ein Modellierungsdilemma: Kompakte Netzspezifikationen und kompakte Netztopologien lassen sich bei der Gestaltung von Netzmodellen nicht gleichzeitig realisieren. Schon während der prädikatenlogischen Objektmodellierung wird die Vorentscheidung darüber getroffen, welcher Antipode der Vorzug gegeben wird.

Die zweite Unterbestimmtheit der Objektmodellierung erstreckt sich auf die Formulierung von Sachverhalten, die Folgerungszusammenhänge darstellen[6)]. Daher liegt eine implikative Unterbestimmtheit vor. Sie beruht auf dem Freiheitsgrad, Folgerungen entweder als subjunktive Klauseln oder aber als Produktionsregeln darzustellen. Im Gegensatz zur attributiven Unterbestimmtheit besitzt die Entscheidung, in welcher Weise dieser Freiheitsgrad bei der Modellgestaltung ausgeschöpft wird, erhebliche theoretische Bedeutung. Denn die Benutzung von subjunktiven Klauseln führt zu einer deklarativen, die Verwendung von Produktionsregeln dagegen zu einer operationalen Objektmodellierung. Beide Modellierungsarten unterscheiden sich wesentlich. Dies wird im folgenden näher erläutert.

Auf den ersten Blick scheinen sich subjunktive Klauseln und Produktionsregeln stark zu ähneln. Denn sie besitzen dieselbe syntaktische Struktur von Subjugaten[7)]. Eine typische Klausel[8)] "kla" und eine typische Produktionsregel "pr" lassen sich notieren als:

$$\text{kla} \Leftrightarrow (\text{prä}_{u(1)} \wedge \ldots \wedge \text{prä}_{u(QN)}) \rightarrow (\text{prä}_{u(QN+1)} \vee \ldots \vee \text{prä}_{u(Q)})$$

$$\text{pr} \Leftrightarrow (\text{con}_1 \wedge \ldots \wedge \text{con}_V) \rightarrow (\text{act}_1 \wedge \ldots \wedge \text{act}_W)$$

Die strukturelle Übereinstimmung zwischen subjunktiven Klauseln und Produktionsregeln bleibt jedoch auf den Aspekt ihrer subjugatartigen Formelgestalt beschränkt. Davon abgesehen klaffen zwischen Klauseln und Produktionsregeln erhebliche Unterschiede:

- Bereits auf syntaktischer Ebene können sich subjugatartige Klauseln und Produktionsregeln hinsichtlich der Verknüpfungslogik ihrer Konklusionen unterscheiden. Dies ist genau dann der Fall, wenn sie aus mehreren positiven Literalen $\text{prä}_{u(q)}$ mit $q \in \{Q_{N+1},\ldots,Q\}$ bzw. Wirkungsformeln act_w mit $w \in \{1,\ldots,W\}$ zusammengesetzt sind. Denn die positiven Literale formen ein Adjugat, die Wirkungsformeln dagegen ein Konjugat.

- Die negativen Literale $\text{prä}_{u(q)}$ der subjunktiven Klausel "kla" mit $q \in \{1,\ldots,Q_N\}$ entsprechen keineswegs den positionsgleichen Voraussetzungsformeln con_v der Produktionsregel "pr" mit $v \in \{1,\ldots,V\}$. Vielmehr finden sich die negativen Literale $\text{prä}_{u(q)}$ in der Produktionsregel als Wirkungsformeln act_w wieder, sofern es sich dabei um Reduktionsformeln handelt. Denn negative Literale $\text{prä}_{u(q)}$ und Reduktionsformeln red_w stimmen darin überein, in einem Netzmodell jeweils zu einer Eingangsstelle derjenigen Transition zu führen, welche die betrachtete Klausel oder Produktionsregel repräsentiert. Daher besitzen die stellenartigen Netzäquivalente von negativen Literalen und Reduktionsformeln die gleiche operationale Semantik: Beide werden hinsichtlich ihrer aktuellen Markierung beim Schalten der vorgenannten Transition vermindert.

- Die positiven Literale $\text{prä}_{u(q)}$ der subjunktiven Klausel "kla" mit $q \in \{Q_{N+1},\ldots,Q\}$ fallen mit den positionsgleichen Wirkungsformeln act_w der Produktionsregel "pr" nicht notwendig zusammen. Statt dessen entsprechen die positiven Literale $\text{prä}_{u(q)}$ in der Produktionsregel nur denjenigen Wirkungsformeln act_w, die Expansionsformeln darstellen. Denn positive Literale $\text{prä}_{u(q)}$ und Expansionsformeln exp_w stimmen darin überein, in einem Netzmodell jeweils zu einer Ausgangsstelle derjenigen Transition zu führen, welche die betrachtete Klausel oder Produktionsregel repräsentiert. Deshalb weisen nur die stellenartigen Netzäquivalente von positiven Literalen und Expansionsformeln die gleiche operationale Semantik auf: In beiden Fällen wird ihre aktuelle Markierung beim Schalten der vorgenannten Transition vergrößert.

- Die Entsprechungen zwischen negativen und positiven Literalen einerseits sowie Reduktions- bzw. Expansionsformeln andererseits gelten jedoch nur hinsichtlich der gleichartigen Schaltwirkung, die Transitionen im Netzmodell auf die literal- und formelrepräsentierenden Stellen ausüben. Davon abgesehen unterscheiden sich die vorgenannten Literale und Wirkungsformeln erheblich. Denn Klauseln und ihre Literale gehören der klassischen Prädikatenlogik an, in deren deklarativer Semantik Veränderungen von formelzugehörigen Faktenmengen nicht unmittelbar dargestellt werden. Produktionsregeln und ihre Wirkungsformeln betonen dagegen den Aspekt der operationalen Semantik, indem sie die Veränderungen von Faktenmengen unmittelbar ausdrücken.

- Die Voraussetzungsformeln con_v der Produktionsregel "pr" finden in der Klausel "kla" überhaupt keine direkte Entsprechung. Dies gilt insbesondere für die algebraischen Bedingungsformeln, die in den Voraussetzungsformeln der Produktionsregel enthalten sein können[9)].

Die Modellierung von Folgerungszusammenhängen durch Klauseln oder Produktionsregeln verhalten sich aufgrund der vorgenannten Unterschiede nur so lange syntaktisch gleichwertig, wie relativ einfach strukturierte Folgerungen formuliert werden. Für solche Fälle läßt sich zeigen, daß subjunktive Klauseln und Produktionsregeln unmittelbar ineinander transformiert werden können[10)]. Anspruchsvoller strukturierte Folgerungszusammenhänge führen dagegen dazu, daß sich subjunktive Klauseln und Produktionsregeln noch nicht einmal auf der syntaktischen Ebene unmittelbar entsprechen. Diese Unterschiedlichkeit von Produktionsregeln und subjunktiven Klauseln wird jedoch kaum thematisiert[11)].

Die syntaktische Verschiedenartigkeit von Klauseln und Produktionsregeln betrifft erstens die explizite Berücksichtigung algebraischer Ausdrucksmöglichkeiten. Sie erfolgt nur seitens der Produktionsregeln durch ihre Bedingungs- und Bestimmungsformeln. Zwar ist es keineswegs ausgeschlossen, daß die atomaren Formeln einer Klausel ebenfalls algebraische Formeln darstellen[12)]. Doch erfolgt die Einbindung algebraischer Ausdrucksmittel in der Produktionsregeldefinition wesentlich transparenter. Daher bieten sich in erster Linie Produktionsregeln an, um in prädikatenlogische Objektmodelle das algebraische Ausdruckspotential des Signaturkonzepts einzubeziehen.

Ein zweiter syntaktischer Unterschied erstreckt sich auf typische Folgerungszusammenhänge[13)]. Ein Folgerungszusammenhang wird hier als typisch bezeichnet, falls seine Prämisse und seine Konklusion jeweils ein Konjugat aus mehreren atomaren Formelvorkommnissen darstellen[14)]. In dieser Hinsicht weichen Produktionsregeln und Klauseln durch die kon- bzw. adjunktive Verknüpfungslogik ihrer Konklusionsformeln erheblich voneinander ab. Daraus folgen zwei wesentliche Konsequenzen.

Einerseits kann ein typischer Folgerungszusammenhang zwar durch genau eine Produktionsregel modelliert werden. Denn ihre konjunktive Verknüpfung von Voraussetzungs- und Wirkungsformeln entspricht genau den Konjugaten aus der Prämisse und der Konklusion des Folgerungszusammenhangs. Aber es ist unmöglich, den gleichen Folgerungszusammenhang durch nur eine subjunktiv formulierte Klausel auszudrücken. Denn die adjunktive Verknüpfung der positiven Literale einer Klausel widerspricht dem Konjugat aus der Konklusion des Folgerungszusammenhangs. Daher müssen mehrere Klauseln mit jeweils einem positiven Literal verwendet werden, um den Folgerungszusammenhang abzubilden. Die Klauseln gelten dann - zumindest implizit - als konjunktiv verknüpft. Folglich wird der Folgerungszusammenhang in einem Netzmodell entweder auf genau eine produktionsregelrepräsentierende Transition oder aber auf mehrere klauselrepräsentierende Transitionen abgebildet. Daher ist die Transitionenanzahl unterbestimmt, die in einem Netzmodell für die Repräsentation des Folgerungszusammenhangs aus dem Objektmodell erforderlich ist. Abb. 54 u. 55 auf der nächsten Seite verdeutlichen diese Verschiedenartigkeit von Klauseln und Produktionsregeln anhand eines einfachen Beispiels. Es wird ein typischer Folgerungszusammenhang betrachtet, der aus zwei atomaren Formeln $prä_1$ und $prä_2$ zwei andere atomare Formeln $prä_3$ und $prä_4$ folgert[15)]. Da seine Konklusion ein Konjugat aus zwei atomaren Formeln darstellt, werden auch *zwei* klauselrepräsentierende Transitionen t_1 und t_2 benötigt[16)]. Dagegen reicht für die Darstellung desselben Folgerungszusammenhangs *eine* produktionsregelrepräsentierende Transition aus[17)]. Dabei handelt es sich um eine typische Produktionsregel, welche die Gültigkeit der beiden Antezedensformeln $prä_1$ und $prä_2$ nicht aufhebt. Die

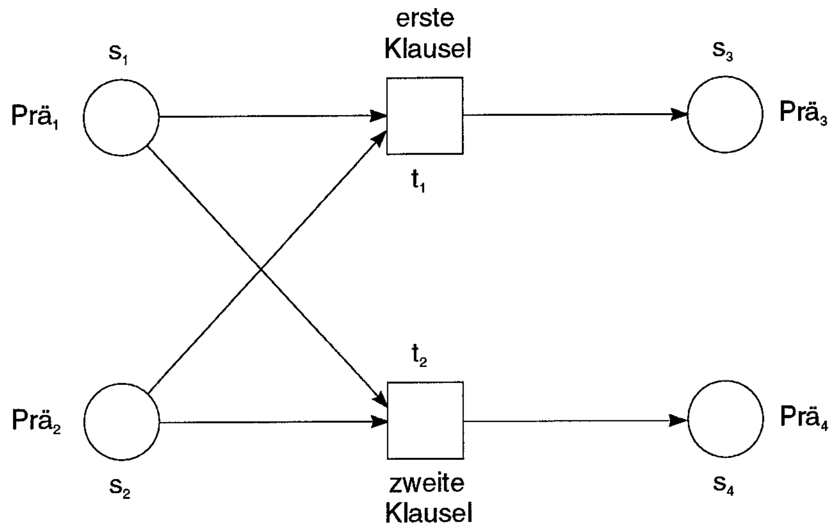

Abb. 54: Deklarative Repräsentation eines typischen Folgerungszusammenhangs durch zwei Klauseln

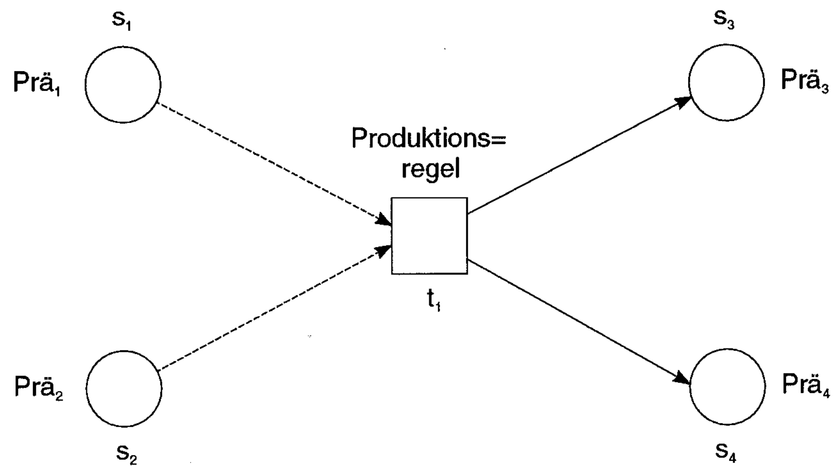

Abb. 55: Operationale Repräsentation eines typischen Folgerungszusammenhangs durch eine Produktionsregel

regelrepräsentierende Transition t_3 gereift daher über zwei Informationskanten auf die beiden formelrepräsentierenden Stellen s_1 und s_2 zu.

Andererseits besitzt eine Transition in einem Netzmodell mit mehreren Ausgangsstellen zwei unterschiedliche logische Interpretationen. Die Interpretation richtet sich danach, ob die Transition entweder als Produktionsregel oder aber als Klausel aufgefaßt wird: Falls die Transition eine Produktionsregel repräsentiert, sind die Prädikatssymbole, die ihren Ausgangsstellen zugeordnet sind, konjunktiv verknüpft. Andernfalls - wenn sie eine Klausel darstellt - gelten die zugeordneten Prädikatssymbole als adjunktiv verknüpft. Folglich bleibt die Verknüpfungslogik der Prädikatssymbole aus dem mehrstelligen Nachbereich einer Transition unterbestimmt, solange nicht bekannt ist, ob sie entweder eine Produktionsregel oder aber eine Klausel repräsentiert.

Die implikative Unterbestimmtheit der Objektmodellierung führt dazu, daß deklarative und operationale Netzmodelle auseinanderfallen können, obwohl sie das gleiche Modellierungsobjekt abbilden. Dazu tragen zunächst die beiden voranstehend erläuterten syntaktischen Aspekte bei, die sich auf unterschiedliche Transitionenanzahlen und verschiedenartige Verknüpfungslogiken von Transitionen mit mehrere Ausgangsstellen erstrecken. Darüber hinaus wird von der implikativen Unterbestimmtheit aber auch die Semantik von Netzmodellen betroffen. Sie fällt bei deklarativen und operationalen Netzmodellen grundsätzlich verschieden aus. Denn das Schaltverhalten und die Markierungen von Netzmodellen besitzen unterschiedliche logische Qualität je nachdem, ob die zugrundeliegenden Objektmodelle mit der Hilfe von Klauseln oder von Produktionsregeln formuliert wurden.

Die logische Verschiedenartigkeit von Produktionsregeln und subjunktiven Klauseln äußert sich bereits darin, daß Formelgültigkeiten und Faktenmengen vollkommen verschieden behandelt werden. Bei Klauseln handelt es sich um rein syntaktisch definierte Formelverknüpfungen. Für sie spielt die Gültigkeit der beteiligten atomaren Formeln zunächst überhaupt keine Rolle[18]. Produktionsregeln besitzen dagegen über ihre syntaktische, subjugatartige Formulierung hinaus eine operationale Semantik: Einerseits müssen alle Formeln, welche die Regelvoraussetzung spezifizieren, gültig sein, damit die Produktionsregel überhaupt ausgeführt werden kann. Andererseits beschreibt die Regelwirkung, in welchem Ausmaß die Faktenmengen von Prädikatssymbolen, die in der Regelwirkung angesprochen werden, durch die Regelausführung verkleinert oder vergrößert werden. Diese Verschiedenartigkeit von Klauseln und Produktionsregeln führt bei der Repräsentation von Folgerungszusammenhängen[19] zu erheblichen Konsequenzen.

Falls ein typischer Folgerungszusammenhang im zugrundeliegenden Objektmodell durch eine Produktionsregel erfaßt wird, entspricht ihm im Netzmodell immer genau eine Transition. Das Schalten dieser Transition kann mit dem Ziehen derjenigen Folgerung identifiziert werden, deren Inhalt die Produktionsregel spezifiziert. Die Netzmarkierungen, die vor und nach dem Schalten der Transition vorliegen, stellen Faktenmengen dar. Sie geben diejenigen atomaren Formelvorkommnisse wieder, die vor bzw. nach dem Ausführen der Schlußfolgerung als gültig bekannt sind. Dabei entspricht jede Marke, die sich auf einer Stelle befindet, genau einem gültigen atomaren Formelvorkommnis. Daher gibt eine Netzmarkierung den jeweils aktuellen Zustand des zugrundeliegenden prädikatenlogischen Objektmodells durch die Gesamtheit aller atomaren Formelvorkommnisse wieder, die in diesem Modellzustand gültig sind[20]. Deswegen gestattet die Verwendung von Produktionsregeln, zwischen operationalen Objektmodellen und ihren zugehörigen Netzmodellen transparente logische Beziehungen herzustellen[21].

Diese übersichtliche Interpretation von Netzmodellen kann aber nicht mehr aufrechterhalten werden, wenn typische Folgerungszusammenhänge in deklarativen Objektmodellen durch subjunktive Klauseln dargestellt werden. Denn *ein* Folgerungszusammenhang muß durch so viele separate subjunktive Klauseln ausgedrückt werden, wie konjunktiv verknüpfte atomare Formeln in seiner Konklusion enthalten sind. Diese Klauseln, die im Objektmodell denselben Folgerungszusammenhang gemeinsam abbilden, werden im deklarativen Netzmodell per constructionem durch jeweils eine Transition repräsentiert. Dann läßt sich im Netzmodell jede der klauselspezifischen Transitionen separat schalten. Jeder dieser Schaltakte gibt aber nur noch einen Ausschnitt aus dem zugrundeliegenden Folgerungszusammenhang wieder. Solche Folgerungsfragmente stellen zwar weiterhin logisch korrekte Folgerungen dar[22]. Aber sie entsprechen nicht mehr dem vollständigen Gehalt des vorausgesetzten typischen Folgerungszusammenhangs.

Folglich kann das Schalten *einzelner Transitionen* in einem deklarativen Netzmodell nicht mehr mit dem Ziehen einer typischen Schlußfolgerung im Objektmodell identifiziert werden. Statt dessen finden sich prädikatenlogische Inferenzen, die im Objektmodell zulässig sind, im deklarativen Netzmodell als *Schaltprozesse* wieder[23]. Allerdings müssen die Schaltprozesse eine recht abstrakt anmutende Bedingung erfüllen: Sie stellen nur dann das Äquivalent von Inferenzprozessen dar, wenn sie die markenfreie Ausgangsmarkierung reproduzieren. Dabei spielen Schaltakte einzelner Transitionen nur noch eine beweistechnische Rolle[24], die im zugrundeliegenden Objektmodell keine unmittelbar anschauliche Entsprechung besitzt.

Fakten eines deklarativen Objektmodells werden im zugehörigen Netzmodell als Teilnetze mit einer speziellen Topologie repräsentiert: Die Teilnetze umfassen jeweils genau ein Transition, die unter der Ausgangsmarkierung aktiviert ist. Die Transition besitzt genau eine Ausgangs-, aber keine Eingangsstelle. Die eine Kante zwischen der Transition und ihrer Ausgangsstelle ist mit derjenigen atomaren Grundtermformel gewichtet, deren Gültigkeit faktisch bekannt ist. Diese Formelgültigkeit wird aber nicht explizit als Markierung der Ausgangsstelle dargestellt. Denn das deklarative Netzmodell besitzt per constructionem eine markenfreie Ausgangsmarkierung. Statt dessen ergibt sich die Gültigkeit der faktischen Formel nur implizit aus der verschärften Gültigkeits-Prämisse für prädikatenlogische Objektmodelle. Ihr zufolge werden alle atomaren Grundtermformeln eines Objektmodells als gültig unterstellt. Aufgrund der voranstehend skizzierten Umstände geht die wechselseitige Entsprechung von Netzmarkierungen und Faktenmengen verloren. Daher besitzen die Netzmarkierungen keine anschauliche Bedeutung mehr. Sie repräsentieren nur noch Zustände von Inferenzprozessen, die in den deklarativen Netzmodellen ausgeführt werden können. Folglich spielen die Markierungen nur noch eine abstrakte inferenztechnische Rolle.

Die voranstehenden Anmerkungen lassen erkennen, daß in einem deklarativen Netzmodell die Schaltakte einzelner Transitionen und die Markierungen von Stellen nicht unmittelbar mit intuitiv anschaulichen Aspekten des zugrundeliegenden prädikatenlogischen Objektmodells verknüpft werden können. Daher besitzen deklarative Netzmodelle eine weniger transparente logische Interpretation als die früher behandelten operationalen Netzmodelle. Dieser Sachverhalt trägt dazu bei, daß der Verf. die Objektmodellierung auf der Basis von Produktionsregeln bevorzugt[25].

Darüber hinaus lassen sich Veränderungen von Faktenmengen in operationalen Objektmodellen durch die Regelwirkungen von Produktionsregeln direkt und komplikationslos abbilden. Daher eignen sich Produktionsregeln in vorzüglicher Weise, um reale Aktionen zu modellieren[26]. Dabei geben die Produktionsregeln an, wie sich durch die Aktionsausführungen die Wahrheitswerte von realitätsbeschreibenden Formeln im Objektmodell verändern. Es könnte zwar der Eindruck bestehen, subjunktive Klauseln ließen sich aufgrund ihrer ähnlichen syntaktischen Subjugatstruktur ebenfalls heranziehen, um das Wirkungsgefüge von Aktionen zu beschreiben. Dies bereitete aber erhebliche Schwierigkeiten, weil diesen Subjugaten eine explizit definierte operationale Semantik fehlt. Insbesondere gestatten sie nicht, die aktionsbedingten Veränderungen von Faktenmengen unmittelbar auszudrücken[27]. Produktionsregeln bieten hierfür jedoch ein wohldefiniertes und transparentes Konzept an. Daher wird aktionsbeschreibendes Wissen in dieser Arbeit von vornherein durch operationale Objektmodellierungen in Produktionsregelform repräsentiert[28].

Allerdings fordern die Vorzüge operationaler Netzmodelle, die hinsichtlich der anschaulichen logischen Modellinterpretation bei der Repräsentation von aktionsbeschreibendem Wissen bestehen, auch einen Preis. Denn die prädikatenlogischen Analyseinstrumente, mit deren Hilfe sich das Inferenzpotential deklarativer Netzmodelle untersuchen läßt[29)], können auf operationale Netzmodelle nicht angewendet werden. Diese Instrumente setzen voraus, daß für alle Transitionen mit mehreren Ausgangsstellen die stellenzugehörigen Prädikatssymbole adjunktiv verknüpft sind[30)]. Daher besteht ein Modellierungsdilemma: Die oben präferierte Repräsentation von Folgerungszusammenhängen durch Produktionsregeln und die Anwendung prädikatenlogischer Analyseinstrumente auf Netzmodelle schließen sich wechselseitig aus[31)].

Auch angesichts dieses Modellierungskonflikts bevorzugt der Verf. weiterhin den Gebrauch von Produktionsregeln für die Abbildung von Folgerungszusammenhängen. Die prädikatenlogischen Analyseinstrumente für Netzmodelle erscheinen zwar aus theoretischer Perspektive hochinteressant[32)]. Aber sie sind für die praktische Untersuchung von Netzmodellen nicht erforderlich. Denn alle Verhaltensaspekte von Netzmodellen lassen sich auch unabhängig von den prädikatenlogischen Analyseinstrumenten mit Hilfe der Erreichbarkeitsanalyse gewinnen. Sie wird später ausführlicher dargestellt[33)].

Anmerkungen zum Kapitel:

1) Die beiden Optionen können auch derart miteinander kombiniert werden, daß manche Objekteigenschaften durch zusammengesetzte Terme und andere 1-stellige Prädikatssymbole erfaßt werden.

2) Es wird hierbei jeweils auf teilevaluierte Terme Bezug genommen.

3) Vgl. dazu den Hinweis, daß in konventionellen Abhandlungen zumeist nur Prädikatssymbole mit flach definierten Argumenten verwendet werden. Dann lassen sich Objekteigenschaften grundsätzlich nur mit der Hilfe von 1-stelligen Prädikatssymbolen ausdrücken. Vgl. beispielsweise diese Formulierungsweise bei BUCHER (1987), S. 134 u. 162ff.

4) Mit Äquivalenz ist hier der strenge logische Äquivalenzbegriff gemeint: Die Menge aller logischen Konsequenzen, die aus den beiden Netzmodellvarianten gezogen werden können, fallen identisch aus. Die logische Konsequenzenmenge von Netzmodellen wird an anderer Stelle näher behandelt.

5) Bezugspunkt ist die alternative Verwendung 1-stelliger Prädikatssymbole.

6) Als ein Folgerungszusammenhang wird hier jeder Sachverhalt verstanden, der sich natürlichsprachlich als eine "Wenn..., dann..."-Beziehung umschreiben läßt. Die "Wenn..."-Komponente eines Folgerungszusammenhangs wird nachfolgend als Prämisse thematisiert, während die "dann..."-Komponente als Konklusion oder Folgerungsergebnis angesprochen wird.

7) Daher kann einem prädikatenlogischen Objektmodell oftmals nicht sofort angesehen werden, ob es sich um ein deklaratives Modell, das sich auf Klauseln stützt, oder um ein operationales Modell handelt, das aus Produktionsregeln aufgebaut ist. Auch die Modellgestalter legen sich zumeist nicht darauf fest, ob sie vom allgemeinen prädikatenlogischen Konzept der konjunktiven Normalformen und Klauseln oder vom speziellen Ansatz der Produktionsregelsysteme ausgegangen sind. Entsprechend bleibt im Dunkeln, ob die objektmodellierenden Formelsysteme einen deklarativen oder operationalen Charakter besitzen sollen. Vgl. dazu beispielsweise die Ausführungen von VAGIN (1988), S. 100ff., die sich sowohl in den Klausel- als auch in den Produktionsregelkontext einordnen lassen. Vgl. ebenso die einleitende Anmerkung zur Vermengung von operationalen und deklarativen Objektmodellen bei der Repräsentation von Produktionsregeln.
Die Gefahr, deklarative und operationale Objektmodellierungen miteinander zu verquicken, braucht keineswegs immer zu drohen. Denn es besteht kein Zwang, alle Klauseln, die aus mindestens einem negativen und mindestens einem positiven Literal bestehen, in der nachfolgend thematisierten Subjugatform auszudrücken. Die anschließenden Erörterungen gelten also nur in dem Ausmaß, wie diese Klauseldarstellungsweise bei der Objektmodellierung tatsächlich benutzt wird, um Folgerungszusammenhänge auszudrücken.

8) Die subjunktiven Klauseln werden vereinfacht auch nur als Klauseln angesprochen.

9) Darüber hinaus finden auch die algebraischen Bestimmungsformeln, die in den Wirkungsformeln der Produktionsregel vorkommen können, in der Klausel "kla" kein Pendant.

10) Die Gleichwertigkeit von subjunktiv dargestellten Klauseln einerseits und Produktionsregeln andererseits läßt sich garantieren, falls zwei Anforderungen erfüllt sind. Einerseits muß es sich bei den Klauseln um HORN-Klauseln handeln, die jeweils genau ein positives Literal besitzen. Andererseits müssen die Produktionsregeln folgende Einschränkungen erfüllen: Algebraische Bedingungs- und Bestimmungsformeln kommen in den Produktionsregeln nicht vor, sofern es sich nicht um die Voreinstellung der tautologischen Formel handelt. Die Regelwirkung besteht aus beliebig vielen Reduktions-, aber aus nur genau einer Expansionsformel. Unter diesen speziellen Voraussetzungen ist die Produktionsregel äquivalent mit einer subjunktiv dargestellten HORN-Klausel: Jede Reduktionsformel der Produktionsregel entspricht eineindeutig einem negativen Literal (einer Antezedensformel) der Klausel. Die eine Wirkungsformel der Produktionsregel korrespondiert eineindeutig mit dem einen positiven Literal (der einen Konklusionsformel) der HORN-Klausel. Die beiden tautologischen algebraischen Formeln beeinflussen die Ausführung der Produktionsregel in keiner Weise. Daher ist es unerheblich, daß ihnen kein Teilkonstrukt der äquivalenten Klausel entspricht.

11) Statt dessen dominiert die Strategie, sich von vornherein auf die vorgenannten relativ einfach strukturierten Folgerungszusammenhänge zu beschränken, in denen Produktionsregeln und subjunktive Klauseln weitgehend zusammenfallen. Es ist dagegen außerordentlich schwierig, in prädikatenlogisch ausgerichteten Beiträgen die Behandlung von Folgerungszusammenhängen aufzufinden, die zu unterschiedlichen deklarativen und operationalen Objektmodellen führen würden. Dies wäre aufgrund der voranstehenden Anmerkung insbesondere dann der Fall, wenn die Folgerungszusammenhänge u.a. durch mehrere konjunktiv verknüpfte Konklusionsformeln beschrieben würden.
Beispielsweise beschäftigt sich der - ansonsten profund und subtil ausgearbeitete - Beitrag von MURATA,TA. (1988b), S. 482ff., nur mit HORN-Klauseln. Jede HORN-Klausel besitzt genau eine Konklusionsformel, wenn sie eine Schlußfolgerungen ausdrückt. Sie wird dann durch eine Transition mit genau einer Ausgangsstelle repräsentiert: "An element $t \in T$ defines a logical implication between its *input places* (predicates) and its *output place* (predicate)."

(S. 484; kursive Hervorhebung durch den Verf.). Dieser Einschränkung auf HORN-Klauseln wird die sachlich falsche Behauptung zugrundegelegt (S. 482), jedes logisch formulierbare Problem könne ausschließlich mit der Hilfe von HORN-Klauseln ausgedrückt werden. Die Unmöglichkeit, alle prädikatenlogischen Formeln als HORN-Klauseln zu reformulieren, wurde jedoch bereits in einer früheren Anmerkung dargelegt.

12) Es wurde schon an früherer Stelle vereinbart, die Prädikatenlogik in dieser Arbeit stets so weit aufzufassen, daß sie auch den algebraischen Formulierungsreichtum umfaßt.

13) Alle anschließenden Ausführungen beziehen sich auf typische Folgerungszusammenhänge, auch wenn das Attribut "typisch" nicht ausdrücklich verwendet wird. Auf untypische Folgerungszusammenhänge wird explizit hingewiesen.

14) Vgl. dazu die entsprechende Formierung einer typischen Klausel und einer typischen Produktionsregel. Alle drei typischen Konstrukte enthalten weder Negate noch Adjugate von Formelvorkommnissen in ihren subjunktiven Darstellungen.

15) Den formelrepräsentierenden Stellen werden in Synthetischen Netzen stets die Namen derjenigen Prädikatssymbole zugeordnet, aus denen die Formeln abgeleitet sind. Daher sind in den beiden nachfolgenden Abbildungen die Stellen nicht mit den Formeln $prä_i(...)$ mit $i \in \{1,2,3,4\}$, sondern mit den Namen $Prä_i$ der zugrundeliegenden Prädikatssymbole beschriftet. Darüber hinaus wird hier der Übersichtlichkeit halber von den Formelargumenten "(...)" abstrahiert. Daher werden nur die Formelnamen "$prä_i$" angeführt.

16) Die beiden klauselrepräsentierenden Transitionen gehen aus einer trivialen Äquivalenzformulierung des zugrundeliegenden Folgerungszusammenhangs hervor:

$$(prä_1 \wedge prä_2) \rightarrow (prä_3 \wedge prä_4)$$

$$\Leftrightarrow \quad \neg(prä_1 \wedge prä_2) \vee (prä_3 \wedge prä_4)$$

$$\Leftrightarrow \quad (\neg prä_1) \vee (\neg prä_2) \vee \ (prä_3 \ prä_4)$$

$$\Leftrightarrow \quad ((\neg prä_1) \vee (\neg prä_2) \vee prä_3) \wedge ((\neg prä_1) \vee (\neg prä_2) \vee prä_4)$$

Die beiden zuletzt angeführten Adjugate stellen jeweils eine (HORN-)Klausel mit genau einem positiven Literal dar.

17) Dabei wird angenommen, daß die beiden Formeln $prä_1$ und $prä_2$ aus der Prämisse des Folgerungszusammenhangs ihre Gültigkeit nicht verlieren, wenn die Konklusion der Formeln $prä_3$ und $prä_4$ gezogen wird. Daher sind die beiden Stellen, welche die Prämissenformeln repräsentieren, mit der produktionsregeldarstellenden Transition durch Informationskanten verknüpft.

18) Fakten, die Kenntnisse über gültige Formeln ausdrücken, werden nur mittels spezieller Teilnetzkonstruktionen implizit erfaßt. Darauf wird noch zurückgekommen.

19) Gemeint sind nachfolgend nur solche Folgerungszusammenhänge, deren Folgerungsergebnisse jeweils durch ein Konjugat aus mehreren atomaren Formeln spezifiziert werden.

20) Diese faktenbezogene Markierungsdeutung lag auch schon der früheren Entfaltung Synthetischer Netze zugrunde.

21) Die logische Beziehung zwischen einem operationalen Objektmodell und seinem zugehörigen Netzmodell läßt sich noch weiter ausbauen. Ausgangspunkt ist einerseits die frühere Feststellung, daß die Gesamtheit aller erreichbaren Markierungen eines Netzmodells, das aus einem operationalen Objektmodell abgeleitet ist, mit dem impliziten Wissen über alle zulässigen Zustände des Objektmodells korrespondiert. Andererseits wird ein Wissenszusammenhang, der aus einer nicht-leeren Formelmenge zusammen mit allen ihren impliziten logischen Konsequenzen besteht, im allgemeinen als eine Theorie bezeichnet. Vgl. zu diesem konventionellen Theorieverständnis ("statement view") z.B. HINST (1983), S. 195.
Aus der Kombination beider voranstehenden Aspekte ergibt sich: Ein Netzmodell besitzt aufgrund seiner Erreichbarkeitsmenge die logische Qualität einer Theorie. Diese Theorie befaßt sich mit der Zulässigkeit von Zuständen des zugrundeliegenden operationalen Objektmodells. MARTI-OLIET und MESEGUER haben diesen Sachverhalt in der plakativen Kapitelüberschrift "Petri nets as theories" zum Ausdruck gebracht (MARTI-OLIET (1989), S. 321). In MARTI-OLIET (1989), S. 321ff. u. 332ff., wird die Identifizierung von Netzen mit Theorien vertieft. Dabei werden Schaltakte von Transitionen, die neue erreichbare Markierungen hervorbringen, mit Ableitungsschritten gleichgesetzt, die in formalen Beweissystemen zum Aufdecken logischer Konsequenzen ausgeführt werden (insbesondere S. 323).

22) Die logische Korrektheit beruht auf der Inferenzregel der Konjunktionsbeseitigung: Wenn eine Konjunktion von (atomaren) Formeln (aus der Konklusion eines Folgerungszusammenhangs) zutrifft, dann trifft auch jede einzelne

der konjunktiv verknüpften Formeln (aus der Folgerungskonklusion) zu. Genau diese Gültigkeit einer einzelnen Konklusionsformel resultiert aus dem Schalten einer Transition aus einem der o.a. Folgerungsfragmente. Vgl. dazu auch die Abb. 54. Dort entsprechen die beiden Transitionen t_1 und t_2 jeweils einem Folgerungsfragment: Das Schalten der ersten (zweiten) Transition korrespondiert mit der Folgerung der Gültigkeit von Formel prä$_3$ (prä$_4$). Vgl. auch zur Inferenzregel der Konjunktionsbeseitigung ESSER,H. (1977a), S. 37a, Fall 16 und 17.

23) Auf die Möglichkeit, durch Schaltprozesse in deklarativen Netzmodellen das Inferenzpotential der zugrundeliegenden Objektmodelle zu untersuchen, wird später in einem Ausblick näher zurückgekommen.

24) Eine subtile Interpretation der logischen Qualität von Schaltprozessen in deklarativen Netzmodellen findet sich bei MURATA,TA. (1988b), S. 486ff. Dabei werden die Schaltprozesse mit prädikatenlogischen Inferenzprozeduren identifiziert, die das kombinierte Unifizierungs- und Resolutionskonzept realisieren. In solchen Schaltprozessen können die Schaltakte von Transitionen drei unterschiedliche Funktionen erfüllen: Sie entsprechen der Bezugnahme auf ein Fakt, wenn die geschalteten Transitionen zu faktrepräsentierenden Teilnetzen gehören. Die Schaltakte bedeuten das Ausführen einer Resolutionsoperation einschließlich der erforderlichen Variablenunifizierungen, falls die betroffenen Transitionen subjunktive Klauseln repräsentieren. Schließlich entsprechen die Schaltakte dem Ableiten der Leerklausel "∅", wenn die geschaltete Transition eine Klausel repräsentiert, die aus genau einer negierten atomaren Formel besteht (Zielklausel). Diese Zusammenhänge lassen sich besonders deutlich aus dem Beispiel von MURATA,TA. (1988b), S. 487, entnehmen. Vgl. darüber hinaus die Anmerkungen zum Netztheorem von MURATA und ZHANG.

25) Dies gilt allerdings nur unter dem Vorbehalt, daß dem Ausdrucksvermögen eines Modellierungskonzepts größeres Gewicht zugemessen wird als seinem Analysepotential. Diese Gewichtung wurde an früherer Stelle für die hier vorgelegte Ausarbeitung gerechtfertigt. Bei einer Bevorzugung des Analyseaspekts ergibt sich hingegen ein anderes Bild: Die Möglichkeiten, das prädikatenlogische Inferenzpotential eines Netzmodells zu erforschen, eröffnet sich erst durch die beiden Netztheoreme, die später skizziert werden. Diese Theoreme sind jedoch nur für deklarative Netzmodelle definiert.

26) Dieser Sachverhalt wird auch aus logischer Richtung unterstrichen: Das aktionslogische Konzept der Linearen Beweise, das von BIBEL entwickelt und bereits in einer früheren Anmerkung angesprochen wurde, repräsentiert Aktionen ebenfalls durch Konstrukte, die den hier bevorzugten Produktionsregeln entsprechen. Denn auch diese Konstrukte stellen Regeln ("rules") in Subjugatform dar. Ihre Konklusionen sind entweder atomare prädikatenlogischen Formeln oder aber *Konjugate* aus solchen Formeln. Vgl. zur konjunktiven Formelverknüpfung in den Konklusionen der aktionsbeschreibenden Regeln z.B. BIBEL (1989), S. 50 u. 52.

27) Es wird hier nicht weiter untersucht, ob Hilfskonstruktionen existieren, die für Klauseln eine zwar nur mittelbare, aber letztlich doch befriedigende Repräsentation von zustandsverändernden Aktionen zulassen. Denn Produktionsregeln leisten diese Aufgabe bereits vorzüglich.

28) Die Möglichkeiten, in Netzmodellen aktionsbeschreibendes Wissen zu repräsentieren, wird noch nicht einmal von Produktionsregeln voll ausgeschöpft. Statt dessen erlaubt es das allgemeine Übergangsschema ÜS, das für Synthetische Netze eingeführt wurde, zustandsverändernde Aktionen noch reichhaltiger zu formulieren, als es in den Produktionsregeln vorgesehen ist. Beispielsweise können im Übergangsschema Haupttestbedingungen für Multimengen atomarer Formelvorkommnisse definiert werden, welche die *Eingangs*kante einer Transition beschriften. Diese Haupttestbedingungen stellen prädikatenlogische Formeln dar, die den algebraischen Bedingungsformeln bed$_{y.v}$ aus der Regelvoraussetzung einer Produktionsregel pr$_y$ ähneln. Jene Bedingungsformeln beziehen sich aber immer auf die Terme in den Argumenten atomarer Formelvorkommnisse aus solchen Multimengen, die eine *Informations*kante der regelrepräsentierenden Transition beschriften. Da Vor- und Informationsbereich jeder Transition disjunkt definiert sind, ist es unmöglich, die o.a. Haupttestbedingungen des Übergangsschemas ÜS durch die Bedingungsformeln von Produktionsregeln wiederzugeben. (Andere Komponenten von Produktionsregeln sind hierzu noch viel weniger in der Lage.) Des weiteren wird in der Produktionsregeldefinition vorausgesetzt, daß die Regelvoraussetzung nur konjunktiv verknüpfte atomare Formelvorkommnisse enthält. Es ist daher unmöglich, dort die Gültigkeit *negierter* Formelvorkommnisse für die Ausführung einer Produktionsregel vorauszusetzen. Das Übergangsschema ÜS kann solche Voraussetzungen durchaus mit Hilfe seiner Prätestbedingungen ausdrücken.
Die voranstehenden Beispiele verdeutlichen, daß das allgemeine Übergangsschema grundsätzlich eine größere Ausdruckskraft besitzt als die Produktionsregeldefinition. Allerdings geht das Übergangsschema über das Formulierungspotential der Prädikatenlogik 1. Stufe - einschließlich ihrer Bereicherungen um algebraisches Signaturkonzept und Multimengen - nicht hinaus. Daraus folgt, daß der Ausdrucksreichtum prädikatenlogischer Objektmodelle durch ihre Produktionsregelform nicht vollständig ausgenutzt wird. Dies gilt a fortiori für Objektmodelle in Klauselform, deren Darstellungsmöglichkeiten hinsichtlich der prädikatenlogischen Semantik noch weitaus geringer ausfallen. Denn Klauseln erwiesen sich bei der Repräsentation von Formelgültigkeiten wesentlich ausdrucksschwächer als Produktionsregeln. Folglich bedeuten Objektmodelle in Klausel- oder Standardform keine umfassende Ausschöpfung des prädikatenlogischen Formulierungsreichtums. Sie stellen zwar leistungsfähige, aber keine allgemeingültigen prädikatenlogischen Objektmodellierungen dar.

Daher setzt der Verf. in dieser Arbeit keineswegs voraus, daß alle Netzmodelle auf Objektmodellen in Klausel- oder Produktionsregelform beruhen müssen. Statt dessen werden alle Netzmodelle zugelassen, welche das Definitionsschema der Synthetischen Netze erfüllen. Dabei wird keine Einschränkung getroffen, aus welcher natürlich- oder formalsprachlichen Grundlage diese Netzmodelle hergeleitet worden sind. Die Transformationsmethoden zur Herleitung von Netzmodellen erfüllen nur eine bedingte Funktion: *Falls* deklarative oder operationale Objektmodelle vorliegen, dann können sie mit Hilfe dieser Methoden in Netzmodelle transformiert werden. Die Umkehrung, daß allen Netzmodellen auch deklarative oder operationale Objektmodelle zugrundeliegen müßten, wird dagegen an keiner Stelle ausgesprochen. So eröffnet der oben exemplarisch verdeutlichte zusätzliche Freiheitsgrad des allgemeinen Übergangsschemas ÜS Ausdrucksmöglichkeiten, die in den früher definierten deklarativen und operationalen Objektmodellen noch nicht enthalten sind.
Daher werden für die spätere Konstruktion von Netzmodellen alle Vorgehensweisen zugelassen, die aus einer natürlichsprachlichen Problembeschreibung ein Netzmodell vom Typ der Synthetischen Netze hervorzubringen vermögen. Die Zwischenstufe über ein deklaratives oder operationales Objektmodell wird keineswegs zwingend vorausgesetzt. Diese beiden Arten der Objektmodellierung wurden lediglich eingeführt, um Methoden für die systematische Konstruktion von Netzmodellen in systematischer Form einführen zu können. Darüber hinaus steht jedoch der Freiraum intuitiver Netzmodellierung offen, die an diese systematische Konstruktionsmethodik nicht gebunden ist. Dieser Freiraum läßt sich jedoch auch nachträglich in die systematische Modellkonstruktion einbringen. Dafür eignet sich wiederum vor allem die operationale Objektmodellierung, weil ihr Ausdrucksmittel - die Produktionsregeln - in erstaunlich flexibler Weise an unterschiedliche Modellierungsintentionen anpassen lassen. Es liegt nicht im Erkenntnisinteresse dieser Arbeit, die Spannweite dieser Adaptionsfähigkeit von Produktionsregeln vollständig auszuloten. Aber es werden an späterer Stelle immerhin mehrere Erweiterungen des Produktionsregel-Konzepts vorgestellt. Beispielsweise wird es dann möglich sein, auch solche Produktionsregeln zu formulieren, deren Regelvoraussetzungen negierte atomare Formelvorkommnisse enthalten. Auf diese Weise wirkt das Petrinetz-Konzept auf den Ausdrucksreichtum von Produktionsregelsystemen in befruchtender Weise zurück. Zugleich wird dadurch die Ausbaufähigkeit des operationalen Modellierungsansatzes demonstriert. Entsprechende Erweiterungsmöglichkeiten läßt die deklarative Objektmodellierung auf der Basis von Klauseln hingegen nicht erkennen. Dies bestätigt nochmals die Präferenz des Verf. für die Verwendung von Produktionsregeln.

29) Sie wurden bereits kurz angesprochen. Sie werden später als invariantenorientierte und als erreichbarkeitsbezogene Netztheoreme näher erläutert.

30) Daher gilt für jedes Netzmodell folgende bedingte Transitionsinterpretation, falls die prädikatenlogischen Analyseinstrumente zur Anwendung gelangen sollen: Wenn eine Transition höchstens eine Ausgangsstelle besitzt, dann ist es unerheblich, ob sie von einem Modellierungsträger als Repräsentation einer Produktionsregel oder einer (HORN-)Klausel interpretiert wird. Wenn eine Transition jedoch mehrere Ausgangsstellen aufweist, dann muß sie als die Darstellung einer (allgemeinen) Klausel interpretiert werden. Folglich bilden in diesem zweiten Fall die Prädikatssymbole, die den Ausgangsstellen der Transition zugeordnet sind, notwendig ein Adjugat.

31) Dies gilt allerdings nur, wenn mindestens ein Folgerungszusammenhang vorliegt, dessen Folgerungsergebnis ein Konjugat aus mehreren atomaren Formeln darstellt.

32) Vgl. dazu die Anmerkungen zu den Netztheoremen von LAUTENBACH sowie von MURATA und ZHANG.

33) Sie wird später ausführlich behandelt.

5.1.3.3 Synthetische Netze zwischen Modellkonzeptualisierung und -implementierung

Netzmodelle, die mit der Hilfe von Synthetischen Netzen gestaltet werden, bilden eine Schnittstelle zwischen prädikatenlogischen Formelsystemen als Objektmodellen auf der einen Seite und PROLOG-Implementierungen der Netzmodelle auf der anderen Seite. Auf den ersten Blick könnte es abundant erscheinen, die Synthetischen Netze zwischen prädikatenlogischen Formelsystemen und PROLOG-Programmen anzusiedeln. Denn die Programmiersprache PROLOG zeichnet sich gerade dadurch aus, für die Implementierung prädikatenlogischer Formelsysteme entwickelt zu sein. Die netzartige Schnittstelle zwischen Formelsystemen der Objektmodelle und PROLOG-Programmen erfüllt jedoch zwei bemerkenswerte Funktionen. Die beiden Schnittstellenfunktionen betreffen die Transparenz und die Universalität der Objektmodellierung.

Durch prädikatenlogische Formelsysteme können zwar Modellierungsobjekte präzise und vollständig abgebildet werden. Aber sie stellen relativ[1)] unanschauliche Konstrukte dar. Gleiches gilt für die PROLOG-Programme, mit deren Hilfe sich solche Formelsysteme implementieren lassen[2)]. Dritte, die diese Formelsysteme oder Programme nicht selbst entworfen haben, vermögen sie oftmals nur schwer oder überhaupt nicht zu verstehen. Daher eignen sie sich kaum als Instrumente, mit deren Hilfe sich ein Modellierungsträger Überblick über oder Einsicht in wesentliche Modellaspekte verschaffen könnte. Ebensowenig taugen sie als Kommunikationsmittel, die Teilnehmer eines arbeitsteiligen Modellierungsprozesses benutzen könnten, um sich über ihr gemeinsames Arbeitsobjekt - das Modell eines Realitätsausschnitts - zu verständigen.

Dagegen sind Netzmodelle in der Lage, die Lücke zwischen der präzisen Vollständigkeit von prädikatenlogischen Formelsystemen einerseits sowie den Transparenzbedürfnissen von Modellierungsträgern andererseits zu schließen. Denn Netzmodelle besitzen als Synthetische Netze die notwendige Ausdruckskraft, um prädikatenlogische Formelsysteme äquivalent repräsentieren zu können. Zugleich verleiht die graphische Visualisierung von Synthetischen Netzen den Netzmodellen aber auch jenes Ausmaß an Anschaulichkeit, das sie als Über- und Einblicks- sowie als Kommunikationsmittel qualifiziert[3)]. Daher fördert der Gebrauch von Netzmodellen die Transparenz der Objektmodellierung erheblich, ohne auf die Präzision und Vollständigkeit von prädikatenlogischer Formulierung und Programmierung verzichten zu müssen.

Die zweite Schnittstellenfunktion von Synthetischen Netzen betrifft die Universalität der Objektmodellierung. Sofern auf die Effizienzvorteile der Automatischen Informationsverarbeitung nicht verzichtet werden soll, verhält sich die prädikatenlogische Modellierung leider nicht universell. Denn die dominierende prädikatenlogische Programmiersprache PROLOG ist auf das Ausdrucksvermögen der HORN-Klausel-Syntax beschränkt[4)]. Es wurde schon früher aufgezeigt, daß sich mit HORN-Klauseln keineswegs alle prädikatenlogischen Formelsysteme wiedergeben lassen. Damit geht ein Teil des prädikatenlogischen Formulierungsreichtums verloren, der in den Formelsystemen von prädikatenlogischen Objektmodellen noch ausgeschöpft werden kann.

Diese Universalitätseinbuße entfällt jedoch, wenn zwischen prädikatenlogische Formelsysteme und PROLOG-Programme eine Schnittstelle aus Synthetischen Netzen tritt. Denn in Synthetischen Netzen können einerseits prädikatenlogische Objektmodelle, die *keinen* syntaktischen Einschränkungen unterliegen, äquivalent repräsentiert werden. Andererseits lassen sich aber Synthetische Netze auf der Basis des Softwarepakets PASIPP als PROLOG-Programme implementieren, deren Syntax auf die Verwendung von HORN-Klauseln beschränkt bleibt. Dennoch erfolgt keine Informationsverfälschung. Denn durch einen Implementierungstrick werden die HORN-Klauseln aus PROLOG-Programmen von den allgemeinen Klauseln oder Produktionsregeln aus prädikatenlogischen Objektmodellen abgekoppelt, ohne daß hierdurch Informationen über das modellierte Objekt verloren gingen oder ungewollt erzeugt würden.

Zentraler Ansatzpunkt sind dabei die Transaktions-Klauseln, die früher für die PASIPP-Implementierung von Synthetischen Netzen definiert wurden. Eine solche Transaktions-Klausel ist ein Konstrukt, das genau eine Transition aus dem Netzmodell, ihre benachbarten Stellen und die zugehörigen Netzbeschriftungen in sich vereinigt. Jede dieser Transaktions-Klauseln stellt per constructionem eine HORN-Klausel dar, die sich unmittelbar als Teil eines PROLOG-Programms implementieren läßt. Dennoch wurde im Zusammenhang mit den Transformationen von Objektmodellen in Netzmodelle aufgezeigt, daß jede allgemeine Klausel oder Produktionsregel durch einen Komplex aus genau einer klausel- bzw. regelspezifischen Transition mit ihren inzidenten Stellen und allen zugehörigen Netzbeschriftungen äquivalent dargestellt werden kann. Daraus folgt das überraschende Ergebnis, daß sich jede allgemeine Klausel oder Produktionsregel aus einem prädikatenlogischen Objektmodell vermittels eines dazwischengeschalteten Netzmodells in einem PROLOG-Programm unverfälscht als HORN-Klausel wiederfindet[5)].

Der wesentliche Implementierungstrick des PASIPP-Konzepts besteht dabei darin, daß die *gesamte* Information über die Klausel bzw. Produktionsregel aus dem Objektmodell *nur* in die Antezedensformeln der HORN-Klausel im PROLOG-Programm eingeht. Die eine Konklusionsformel dieser HORN-Klausel enthält nur noch das klausel*un*spezifische Prädikat "fire(...)" für den Schaltakt der zugrundeliegenden Transition[6)]. Durch diese spezielle Klauselkonstruktion wird es möglich, zwei prima facie widersprechende Aspekte zu vereinbaren: In prädikatenlogischen Objektmodellen dürfen die Subjugate, die allgemeine Klauseln oder Produktionsregeln darstellen, aus mehreren Konklusionsformeln bestehen. In den zugehörigen PROLOG-Programmen besitzen dagegen alle transaktionsimplementierenden HORN-Klauseln jeweils nur genau eine Konklusionsformel[7)]. Ermöglicht wird diese entscheidende Reduktion der Konklusionsformelanzahl durch das Instrument der Netzmodelle. Erst ihre Definition läßt es zu, die allgemeinen Klauseln oder Produktionsregeln von Objektmodellen auf das Artefakt "Transaktions-Klausel" abzubilden, das die formale Gestalt einer HORN-Klausel annimmt.

Die Restriktion der HORN-Klausel-Syntax gilt also nur lokal für die Implementierung jeweils einer Transaktions-Klausel. Die globale Formelverknüpfung im prädikatenlogischen Objektmodell wird dadurch jedoch nicht beeinträchtigt. Denn das prädikatenlogische Formelsystem des Objektmodells kann weiterhin die gesamte Ausdrucksmächtigkeit konjunktiver Normalformen mit allgemeinen Klauseln ausschöpfen[8)]. Damit erfüllen Netzmodelle eine herausragende Schnittstellenfunktion: Sie leiten von der universellen Formulierungskraft der prädikatenlogischen Objektmodellierung zum Rationalisierungspotential der PROLOG-gestützten Modellimplementierung über, ohne daß dabei Einbußen hinsichtlich der Modellierungsuniversalität hingenommen werden müßten.

Anmerkungen zum Kapitel:

1) Bezugspunkte sind jeweils graphische Netzrepräsentationen.

2) Vgl. zur Unanschaulichkeit von PROLOG-Programmen insbesondere BOSSI (1989), S. 97.

3) Vgl. dazu die Ausführungen anläßlich der zusammenfassenden Beurteilung des Petrinetz-Konzepts.

4) Vgl. dazu die Quellen, die zur Programmiersprache PROLOG angeführt wurden.

5) Beispielsweise lassen sich in PROLOG-Programmen keine reinen Adjugate von atomaren Formeln direkt darstellen. HORN-Klauseln sind dazu grundsätzlich nicht in der Lage. Negate von atomaren Formeln können nur als Zielklauseln formuliert werden, nicht aber als "negative" Fakten. Sie lassen sich zwar durch das Ausführen eines PROLOG-Programms bestätigen oder widerlegen. Damit stellen die Negate von atomaren Formeln aber nur *Anfragen* dar, die durch Programmausführungen beantwortet werden können. Es ist jedoch unmöglich, diese Zielklauseln zu benutzen, um Wissen über einen modellierten Realitätsausschnitt zu repräsentieren. Dies verbietet ihr Charakter, Fragen auszudrücken, deren Antwort vom Modellierungsträger vor einer Programmausführung noch nicht gewußt wird. In einem deklarativen Netzmodell ist es dagegen ohne Schwierigkeiten möglich, Adjugate oder Negate von atomaren Formeln zu repräsentieren. Für die entsprechenden Klauseln kla_p und die Topologien $TOP(kla_p)$ der klauseldarstellenden Teilnetze gilt hinsichtlich der beiden einfachst möglichen Adjugat- und Negatformeln:

- Der Adjugatformel $kla_p :\Leftrightarrow (prä_{u(p.1)} \vee prä_{u(p.2)})$ entspricht:

  $TOP(prä_{u(p.1)} \vee prä_{u(p.2)}) = (\{s_{m(p.1)}, s_{m(p.1)}\}, \{t_{n(p)}\}; \{(s_{m(p.1)}, t_{n(p)}), (s_{m(p.2)}, t_{n(p)})\})$

- Der Negatformel $kla_p :\Leftrightarrow \neg prä_{u(p)}$ entspricht:

  $TOP(\neg prä_{u(p)}) = (\{s_{m(p)}\}, \{t_{n(p)}\}; \{(s_{m(p)}, t_{n(p)})\})$

Siehe dazu auch die entsprechenden graphischen Teilnetzdarstellungen in der Abb. 53. Jedes dieser Teilnetze (mit nur einer Transition) wird bei der Netzimplementierung in einem PROLOG-Programm als genau eine Transaktions-Klausel dargestellt.

6) Das Prädikatsargument enthält zwar noch die klauselspezifische Liste aller Variablen, die an der Konstitution der Transaktions-Klausel teilnehmen. Doch werden hierdurch lediglich diejenigen Variablen wiederholt, die bereits im Antezedens der HORN-Klausel enthalten sind.

7) Vgl. dazu die Erläuterung von HORN-Klauseln. Relevant für die Implementierung von Transaktions-Klauseln ist hier der Unterfall der Regelklauseln.

8) Gleiches gilt für die Ausdrucksmächtigkeit von Produktionsregeln.

5.1.3.4 Kanal/Instanz-Netze

Kanal/Instanz-Netze[1)] sind Instrumente für die graphische Visualisierung von unterschiedlich ausgereiften Konzeptualisierungen eines Objektmodells bis hin zu seiner Umsetzung in ein präzise definiertes Netzmodell. Wie die voranstehend erläuterten Transformationsmethoden erfüllen auch sie die Schnittstellenfunktion, zwischen Objektmodellierungen in natürlichsprachlicher oder prädikatenlogischer einerseits und Netzmodellen andererseits zu vermitteln. Allerdings liefern Kanal/Instanz-Netze im Gegensatz zu jenen Transformationsmethoden keine systematische Beschreibung des Übergangs von einem Objekt- zu einem äquivalenten Netzmodell. Statt dessen dienen sie dazu, ein natürlichsprachliches oder prädikatenlogisches Objektmodell auf einer beliebigen Stufe des Konzeptualisierungsprozesses zu veranschaulichen.

Transformationsmethoden und Kanal/Instanz-Netze konkurrieren nicht miteinander, sondern ergänzen sich gegenseitig. Beide erstrecken sich auf unterschiedliche Phasen des Modellierungsprozesses. Die Transformationsmethoden betreffen formalsprachliche Objektmodelle, die in ihrer prädikatenlogischen Klausel- oder Produktionsregelform bereits ein großes Formalisierungsausmaß voraussetzen. Durch die Methodenanwendung werden lediglich hochgradig formalisierte prädikatenlogische Objektmodelle in äquivalente Netzmodelle transformiert. Daher beziehen sich die Transformationsmethoden nur auf jene Phase des Modellierungsprozesses, nach deren Abschluß ein ausgearbeitetes Netzmodell vorliegt. Diese Phase zeichnet sich dadurch aus, daß die Repräsentation des jeweils modellierten Realitätsausschnitts bereits vollständig formalisiert erfolgt.

Kanal/Instanz-Netze lassen sich dagegen auf alle vorangehenden Phasen des Modellierungsprozesses anwenden[2)]. Sie repräsentieren den jeweils erreichten Modellierungsstand in der Gestalt eines visualisierten beschrifteten Graphen. Dieser Graph beschreibt die aktuelle Wahrnehmung des Modellierungsobjekts durch den Modellierungsträger. Fortschritte des Modellierungsprozesses, die mit Variationen der aktuellen Objektmodelle verknüpft sind, schlagen sich in entsprechend veränderten Kanal/Instanz-Netzen nieder. Erst am Ende der fortschreitenden Objektmodellierung liegt ein vollständig formalisiertes deklaratives oder operationales Objektmodell vor, das unmittelbar in ein Netzmodell transformiert werden kann.

Damit eignen sich Kanal/Instanz-Netze vorzüglich für die Aufgabe, die Zwischenergebnisse von Modellierungsprozessen zu dokumentieren. Kanal/Instanz-Netze werde daher in dieser Arbeit als Dokumentationsinstrumente verwendet[3)]. Sie veranschaulichen Zwischenstufen eines Modellierungsprozesses, die zwischen der erstmaligen natürlichsprachlichen Beschreibung des Modellierungsobjekts einerseits und seiner Repräsentation durch ein prädikatenlogisches Objektmodell in Klausel- oder Produktionsregelform andererseits liegen. Das prädikatenlogische Objektmodell braucht noch nicht einmal explizit angegeben zu werden, bevor sein äquivalentes Netzmodell vorgelegt wird[4)]. Dadurch erhält die Dokumentation der Zwischenstufen mit der Hilfe von Kanal/Instanz-Netzen ein besonderes Gewicht. Die Kanal/Instanz-Netze bilden dann einen fließenden Übergang[5)] zwischen dem ursprünglich vorgegebenen natürlichsprachlichen Objektmodell und dem intendierten formalsprachlichen Netzmodell.

Kanal/Instanz-Netze erfüllen ihre Schnittstellenfunktion zwischen natürlichsprachlicher Objektbeschreibung und formalsprachlichem Netzmodell durch ihren semiformalen Charakter[6)]: Einerseits besitzt jedes Kanal/Instanz-Netz einen formalen Kern, der ein Allgemeines Netz darstellt[7)]. Die graphische Visualisierung dieses Netzkerns bildet einen gerichteten bipartiten Graphen, der auch die topologischen Strukturen aller anderen Netzklassen veranschaulicht. In dieser Hinsicht wird die Brücke zu den intendierten formalsprachlichen Netzmodellen geschlagen. Andererseits umfaßt jedes Kanal/Instanz-Netz eine Anschriftenmenge und eine Beschriftungsfunktion. Durch die Beschriftungsfunktion werden Konstituenten des Allgemeinen Netzes einzelne Anschriften zugeordnet. Die Anschriftenmenge unterliegt keinen weiteren Einschränkungen. Daher kann sie sowohl natürlich- als auch formalsprachliche Netzanschriften enthalten[8]. Hierdurch wird die besondere Flexibilität von Kanal/Instanz-Netzen begründet. Hinzu kommt, daß sich die Konstituenten eines Kanal/Instanz-Netzes durch entsprechende Anschriften auf vielfältige Weise interpretieren lassen[9].

In frühen Phasen des Modellierungsprozesses werden vornehmlich natürlichsprachliche Anschriftenmengen verwendet, die sich unmittelbar aus der natürlichsprachlichen Objektbeschreibung übernehmen lassen. In dem Ausmaß, in dem die Objektmodellierung um formalsprachliche Konstrukte bereichert wird, erhält auch die Anschriftenmenge einen entsprechend formaleren Charakter[10]. Mit der schrittweisen Formalisierung des Objektmodells korrespondiert eine Folge von zunehmend formaler gestalteten Kanal/Instanz-Netzen[11]. Am Ende kann ein Kanal/Instanz-Netz stehen, das nur noch rein formalsprachliche Anschriften trägt[12]. Im allgemeinen begleitet diesen netzbasierten Formalisierungsprozeß eine gleich gerichtete sukzessive Präzisierung und Verfeinerung der Objektmodellierung[13].

Definition: Kanal/Instanz-Netz

Ein Kanal/Instanz-Netz ist ein geordnetes 5-Tupel KIN=(S,T;F;AS;bes), für das gilt:
- Das Teiltupel (S,T;F) ist ein Allgemeines Netz.
- Die Anschriftenmenge AS ist eine nicht-leere, endliche Menge aus natürlich- oder formalsprachlichen Ausdrücken.
- Die Beschriftungsfunktion bes: $pot_+(S \cup T \cup F) \rightarrow AS$ ist eine partielle oder vollständige Funktion. Sie kann einzelnen Konstituenten des Teiltupels (S,T;F) oder entsprechenden Konstituentengruppen Anschriften aus der Menge AS zuordnen.

<u>Erläuterungen und Ergänzungen zur Kanal/Instanz-Netz-Definition:</u>

a) Für den Netzkern (S,T;F) werden alle Bezeichnungen übernommen, die früher für Allgemeine Netze eingeführt wurden. Gleiches gilt für den bipartiten gerichteten Graphen GR=(KN,KA), der diesen Netzkern mit $KN = S \cup T$ und KA=F repräsentiert.

b) Die Stellen eines Kanal/Instanz-Netzes werden meistens als "Kanäle" bezeichnet. Sie besitzen die Qualität von passiven oder zustandsartigen Komponenten. Passive Komponenten[14] können Objekte aufnehmen und auch wieder abgeben[15]. Kanäle lassen sich aus dieser Perspektive als Objektspeicher betrachten[16]. Zustandsartige Komponenten[17] dienen dagegen dazu, Zustände darzustellen, die das repräsentierte Modellierungsobjekt im Verlauf von Prozessen anzunehmen - und auch wieder einzubüßen - vermag. Sie bieten sich vor allem an, um logische oder sachliche Bedingungen abzubilden, die erfüllt sein müssen, damit Prozesse ausgeführt werden können. Zumeist fallen in den Kanälen die Perspektiven von passiven und zustandsartigen Komponenten zusammen[18]. Der Einfachheit halber wird fortan nur auf passive Komponenten Bezug genommen, wenn nicht ausdrücklich die zustandsorientierte Perspektive hervorgehoben werden soll.

c) Die Transitionen eines Kanal/Instanz-Netzes werden im allgemeinen als "Instanzen" angesprochen. Sie stellen aktive, vorgangsartige oder ereignishafte Komponenten dar. Aktive Komponenten[19] können Operationen ausführen, die auf die Objekte einwirken[20]. Dabei ziehen sie das jeweils betroffene Objekt aus einer ihrer Eingangsstellen ab, wirken auf das Objekt während der Operationsausführung ein und legen es danach in einer ihrer Ausgangsstellen wieder ab. Bei vorgangsorientierten Komponenten handelt es sich um die vorgenannten Operationen selbst. Der Modellierungsträger kann frei wählen, ob er die Bezugnahme auf den operationsausführenden

Aktor oder auf die ausgeführte Operation bevorzugt[21]. Ereignishafte Komponenten stellen dagegen punktförmige Entitäten dar, welche die Operationsausführungen eines Aktors zeitlich begrenzen[22] oder auch aktorunabhängig geschehen können[23]. Anschließend werden nur die aktiven Komponenten explizit angesprochen, sofern nicht ausdrücklich die vorgangs- oder ereignisbezogene Perspektive im Vordergrund stehen soll.

d) Die Einteilung in passive und aktive[24] Komponenten bezieht sich nur auf das jeweils erreichte Detaillierungsniveau der Objektmodellierung. Bei fortschreitender Modellverfeinerung können die Komponenten in Subkomponenten aufgespaltet werden[25]. Aus der bipartiten Charakteristik des Petrinetz-Konzepts folgt zwangsläufig, daß dann eine verfeinerte Komponente stets in aktive und passive Subkomponenten zerlegt wird[26]. Daher besitzt eine abstrakte Komponente unabhängig davon, ob sie als aktive oder passive Komponente erscheint, stets sowohl aktiven als auch passiven Charakter, wenn sie später in entsprechende Subkomponenten aufgespalten wird[27]. Dieser Hybridcharakter ist zwar auf der ursprünglichen Abstraktionsebene noch opak, wird aber im Augenblick der Komponentenverfeinerung offensichtlich.

e) Die Elemente der Flußrelation F stellen in der graphischen Netzrepräsentation gerichtete Kanten dar. Ihre Interpretation hängt davon ab, wie die jeweils verknüpften Stellen und Transitionen des Kanal/Instanz-Netzes aufgefaßt werden[28]. Z.B. können die Stellen als Objektspeicher und die Transitionen als Operationen auf Objekten angesehen werden. In diesem Fall legen die Kanten durch ihre gerichtete Verknüpfung von Stellen und Transitionen fest, wie bei Operationsausführungen Objekte aus Speichern entnommen und dort wieder abgelegt werden.

f) Die Ausdrücke[29] aus der Anschriftenmenge AS unterliegen keinen syntaktischen Einschränkungen. Wie bereits einleitend herausgestellt wurde, kann es sich sowohl um natürlich- als auch um formalsprachliche Ausdrücke handeln. Falls innerhalb derselben Anschriftenmenge beide Ausdrucksarten enthalten sind, werden die natürlichsprachlichen Anschriften - nicht aber ihre formalsprachlichen Pendants - in Anführungszeichen eingeschlossen. Andernfalls wird auf die Kennzeichnung natürlichsprachlicher Ausdrücke durch Anführungszeichen verzichtet.

g) Die Anschriftenmenge kann auch natürliche Zahlen umfassen, mit denen die Stellen beschriftet werden. Diese Stellenanschriften lassen sich als Marken interpretieren, wie sie für Stelle/Transition-Netze eingeführt wurden. Daher ist es durchaus möglich, Kanal/Instanz-Netze zu markieren[30]. Mit Hilfe der Markenkopien lassen sich Zustandsveränderungen und Objektflüsse visualisieren.

h) Die Beschriftungsfunktion "bes" ist bezüglich der Konstituentenmenge $(S \cup T \cup F)$ des Netzkerns (S,T;F) nur dann vollständig, wenn sie jeder Stelle, jeder Transition und jeder Kante mindestens eine Anschrift zuordnet. Dies ist bei den meisten Kanal/Instanz-Netzen jedoch nicht der Fall. Daher handelt es sich in der Regel um eine partielle Beschriftungsfunktion. Mitunter werden die Kanten eines Kanal/Instanz-Netzes auch überhaupt nicht beschriftet[31]. Auf dieses Ausdrucksmittel wird hier aber nicht verzichtet. Dadurch lassen sich z.B. die Objekte, die über die Netzkanten von Stellen abgezogen oder dort abgelegt werden, durch solche Kantenanschriften unmittelbar identifizieren[32].

i) In der Netzliteratur wird nur die Beschriftung einzelner Konstituenten des Netzkerns zugelassen. Statt dessen wird hier der Definitionsbereich der Beschriftungsfunktion "bes" auf alle nicht-leeren Teilmengen der Konstituentenmenge $(S \cup T \cup F)$ ausgeweitet. Durch diesen Definitionsbereich $pot_+(S \cup T \cup F)$ wird es möglich, auch Gruppen von Konstituenten mit einer gemeinsamen Beschriftung zu versehen[33]. Dies entlastet die graphischen Netzrepräsentationen von redundanten Netzbeschriftungen, sofern andernfalls mehrere Konstituenten mit derselben Anschrift versehen werden müßten[34]. Darüber hinaus läßt sich auf diese Weise spezielles Gruppierungswissen in Kanal/Instanz-Netzen repräsentieren. Das gilt vor allem für die Bündelung von Kanten, die gemeinsame Ein- oder Ausgangskanten eines Netzknotens sind, durch eine besondere[35] logische Verknüpfung. Beispielsweise kann für die Ausgangskanten einer Transition eine disjunktive Verknüpfung ausgezeichnet werden, indem die Elemente aus dieser Kantenteilmenge durch eine abstrakte Anschrift für das ausschließliche "oder" verbunden werden.

j) Bei der Visualisierung des Graphen GR=(KN,KA), der den Netzkern (S,T;F) repräsentiert, können die Beschriftungen einzelner Stellen oder Transitionen in die Graphiksymbole für die jeweils betroffenen Netzknoten eingetragen werden. Dies bereitet bei der Beschriftung von Transitionen keine Schwierigkeiten, weil sich die Abmessungen der rechteckigen Graphiksymbole für transitionsartige Netzknoten an beliebige Beschriftungstexte leicht anpassen lassen. Die Beschriftungen von Stellen durch eingetragene Texte können jedoch bei der Verwendung von kreisförmigen Graphiksymbolen für stellenartige Netzknoten schnell zu Symbolabmessungen führen, die den vorhandenen Visualisierungsplatz einer Zeichenfläche sprengen. Daher wird zugelassen, die stellenartige Netzknoten nicht nur durch kreisförmige, sondern ebenso durch kastenförmige Graphiksymbole mit abgerundeten Ecken[36] darzustellen.

k) Weder die Objekte, die in den Stellen gespeichert und über die Kanten bewegt werden können, noch die Operationen, welche die Transitionen auszuführen vermögen, werden durch den Netzkern (S,T;F) definiert. Der Objekt- und Operationscharakter kann aber durch die Anschriften derjenigen Stellen, Kanten und Transitionen inhaltlich verdeutlicht werden, welche die Objekte speichern und fortbewegen bzw. die Operationen ausführen[37]. Dabei dürfen die Objekt- und Operationsbeschreibungen in den zugehörigen Netzanschriften beliebiges Detaillierungs- oder Abstraktionsniveau annehmen.

l) Kanal/Instanz-Netze sind so flexibel definiert, daß sie sich auf ein breites Spektrum natürlich- oder formalsprachlicher Objektmodellierungen anwenden lassen. Zugleich wird auf eine intuitiv verständliche Netzdefinition Wert gelegt: Es ist möglich, Netzkern und Netzanschriften auch ohne tiefere Kenntnisse des Petrinetz-Konzepts nachzuvollziehen. Die graphische Visualisierung des Netzkerns trägt wesentlich zur Anschaulichkeit der Objektmodellierung bei[38]. Um diese Flexibilität und Transparenz der Kanal/Instanz-Netze nicht zu beeinträchtigen, wird grundsätzlich auf die Definition einer Netzdynamik verzichtet[39]. Kanal/Instanz-Netze besitzen daher grundsätzlich keine Schaltregeln[40].

m) Falls sich auf den Stellen eines Kanal/Instanz-Netzes befinden, so existiert jedoch keine starre formale Regulierung des Markenflusses. Dies folgt aus dem Fehlen jeder Schaltregel. Statt dessen bleibt es der Intuition der Modellbenutzer überlassen, im visualisierten Graphen eines Kanal/Instanz-Netzes die Marken entlang der gerichteten Kanten "fließen"[41] zu lassen.

n) Es wird bereits ein Softwarepaket für das automatengestützte Editieren von Kanal/Instanz-Netzen angeboten[42]. Hierdurch läßt sich die Dokumentation des Modellierungsfortschritts in benutzerfreundlicher Weise abwickeln. Der Editor rationalisiert nicht nur die aufwendige Erstellung und Modifizierung von Netzdarstellungen, sondern überprüft auch die Konsistenz der Benutzereingaben. Dabei wird z.B. gewährleistet, daß die Verfeinerung eines Kanal/Instanz-Netzes immer wieder ein Kanal/Instanz-Netz ergibt[43]. Allerdings läßt er keine Beschriftungen von Netzkanten oder von Konstituentengruppen zu.

o) Durch die systematisch und schrittweise betriebene Verfeinerung eines Kanal/Instanz-Netzes resultiert eine schichtenförmige Modellkonstruktion: Je tiefer die Modellierungsschicht liegt, desto detaillierter fällt ihr Repräsentationsniveau aus. Dieser geschichtete Modellaufbau findet zahlreiche Parallelen bei der Gestaltung informationsverarbeitender Systeme[44], seltener auch im Rahmen der Netzplantechnik[45]. In dieser Hinsicht schlägt die sukzessive Netzverfeinerung eine Brücke zu vertrauten Konzepten für die Modellierung komplexer Systeme.

Ein Beispiel für Kanal/Instanz-Netze:

Die prinzipielle Funktionsweise Flexibler Fertigungssysteme[46] wird auf hohem Abstraktionsniveau durch ein Kanal/Instanz-Netz repräsentiert. Es ist in Abb. auf S. 56 dargestellt. Auf ausführliche Erläuterungen wird verzichtet, um die selbsterklärende Qualität von Kanal/Instanz-Netzen deutlich werden zu lassen. Lediglich die Verknüpfung der beiden Kanten, welche die Transition für die "Zuordnungsentscheidungen" mit den beiden Stellen der "Eingangspuffer" verbinden, erscheint erläuterungsbedürftig. Es handelt sich um eine disjunktive Verknüpfung der Ausgangskanten der Transition: Durch die Zuordnungsentscheidungen, die dem Schalten der Transition entsprechen, wird jeweils ein Auftrag über *genau* eine Ausgangskante an *genau* einen Eingangspuffer weitergeleitet.

Abschließend wird die Konstruktion von Netzmodellen, die dieser Arbeit zugrundeliegt, unter Berücksichtigung der zuvor erläuterten Hilfsmittel des Petrinetz-Konzepts zusammengefaßt. Der Modellierungsprozeß setzt an demjenigen Realitätsausschnitt an, der durch eine Modellierungsaufgabe als relevant ausgezeichnet wird. Der Realitätsausschnitt bildet das Modellierungsobjekt, das der Modellierungsträger während der Objektkonzeptualisierung i.e.S. durch ein explizites natürlichsprachliches Objektmodell beschreibt. Die Ausführung des Konzeptualisierungsprozesses i.e.S. wird nicht dargelegt. Es wird also nicht erörtert, wie ein Modellierungsträger die Modellierungsaufgabe interpretiert, dabei ein implizites mentales Objektmodell entwirft und dieses in ein natürlichsprachlich expliziertes Objektmodell übersetzt.

Statt dessen setzen die Ausführungen dieser Arbeit erst mit dem Ergebnis des Konzeptualisierungsprozesses i.e.S. ein. Ausgangspunkt aller Modellierungen ist eine explizit vorliegende natürlichsprachliche Beschreibung des Modellierungsobjekts. Dieses natürlichsprachliche Objektmodell wird während eines Konzeptualisierungsprozesses i.w.S. in ein Netzmodell verwandelt, das die Gestalt eines Synthetischen Netzes annimmt. Je nachdem, wie schwierig dieser Konzeptualisierungsprozeß i.w.S. ausfällt[47], wird das Netzmodell entweder unmittelbar präsentiert oder aber schrittweise hergeleitet. Falls Zwischenschritte eines Konzeptualisierungsprozesses i.w.S. offengelegt werden, so erfolgt die Präsentation der Zwischenergebnisse mit der Hilfe von Kanal/Instanz-Netzen[48].

Für die Modellierung eines Realitätsausschnitts werden also zwei unterschiedliche Netzklassen mit zwei verschiedenartigen Ansprüchen eingesetzt. Solange sich die Objektmodellierung auf die Phase der Konzeptualisierung (i.w.S.) erstreckt, können Zwischenresultate der Objektkonzeptualisierung durch Kanal/Instanz-Netze veranschaulicht werden. Diese Kanal/Instanz-Netze besitzen eine reine Dokumentationsfunktion. Von einem Netzmodell wird dagegen erst dann gesprochen[49], wenn die Konzeptualisierungsphase mit der Vorlage eines Synthetischen Netzes abgeschlossen ist, das den modellierungsrelevanten Realitätsausschnitt abbildet. Dieses Synthetische Netz erfüllt zwar auch eine Dokumentationsfunktion: Es beschreibt das Endergebnis des gesamten Konzeptualisierungsprozesses. Doch darüber hinaus erfüllt es ebenso die weiterführende Funktion der Modellanalyse. Darauf wird anschließend näher eingegangen.

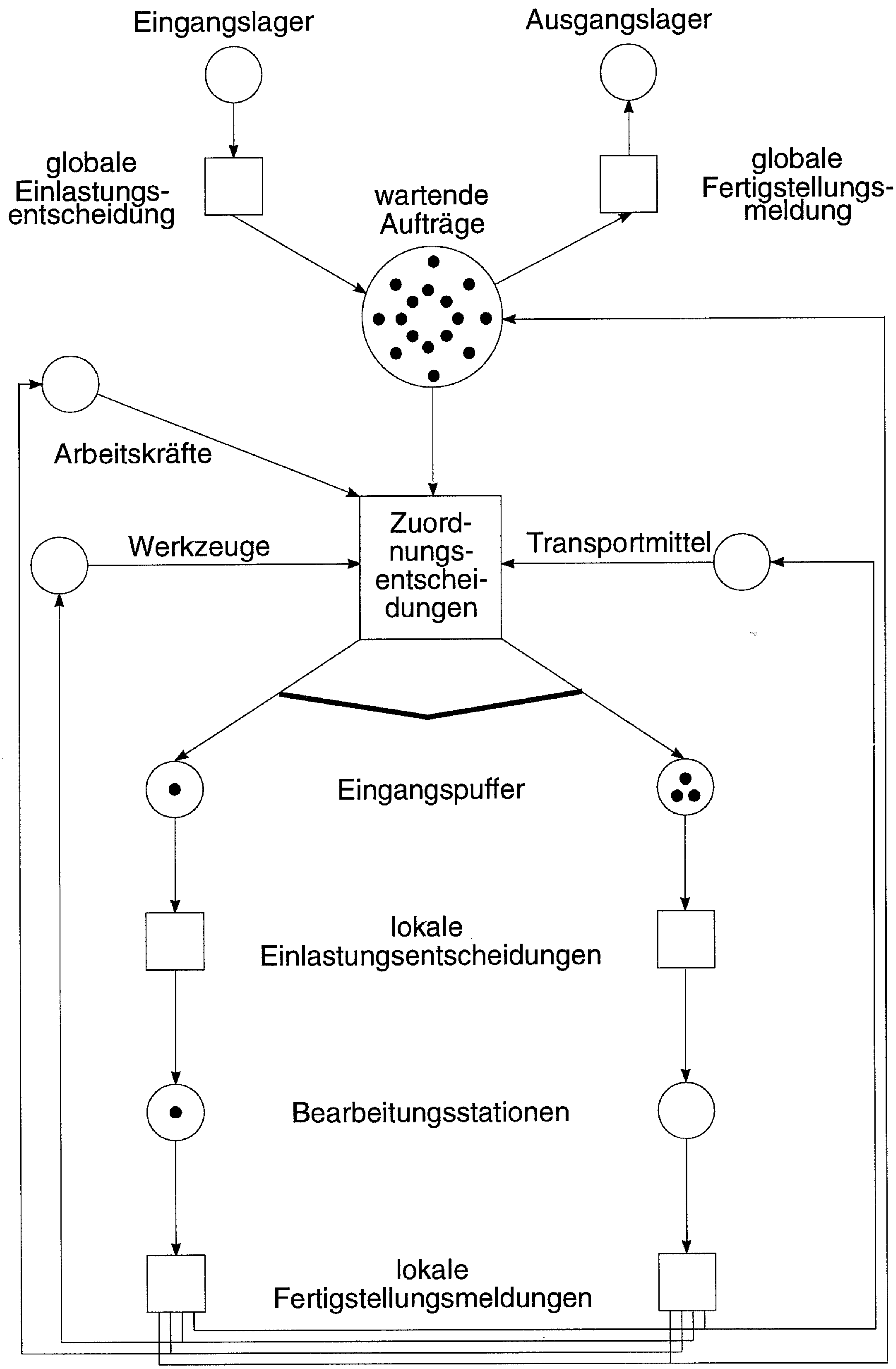

Abb. 56: Kanal/Instanz-Netz für die prinzipielle Funktionsweise eines Flexiblen Fertigungssystems

Anmerkungen zum Kapitel:

1) Vgl. zu Kanal/Instanz-Netzen PETRI,C. (1979c), S. 83; REISIG (1983a), S. 310ff.; RICHTER,G. (1983b), S. 212ff.; RICHTER,G. (1985b), S. 8ff.; IGEL (1986b), S. 2ff.; RICHTER,G. (1988b), S. 123ff., insbesondere 136ff.; DITTRICH,G. (1989b), S. 2ff.; IGEL (1989a), S. 175 u. 192f.; ROSENSTENGEL (1991), S. 53f. u. 116ff.; SCHEER (1991d), S. 131f.
Eng verwandte Spielarten von Petrinetzen behandeln WENDT (1989) S. 217ff.; FEHLING (1990a), S. 6ff. Zwar reden die vorgenannten Autoren nicht explizit von Kanal/Instanz-Netzen. Doch unterscheiden sich ihre Netzvarianten von Kanal/Instanz-Netzen nur unwesentlich. Daher werden die zuvor erwähnten Netzvarianten im folgenden zu Kanal/ Instanz-Netzen im weit gefaßten Sinne gerechnet. Beispielsweise bezieht sich WENDT (1989), S. 217, auf "Instanzennetze". Schon kurz darauf (S. 218) identifiziert er als ihre Hauptkomponenten "Instanzen" und "Speicher". Deren Übereinstimmung mit den Instanzen und Kanälen aus Kanal/Instanz-Netzen ist offensichtlich.
Eine erstaunliche konzeptionelle Annäherung an Kanal/Instanz-Netze findet sich auch bei KNÖLL (1989), S. 49ff. Er argumentiert zwar nicht im Rahmen des Petrinetz-Konzepts. Dennoch verwendet er bipartite Graphen, die in ihrer visualisierten Darstellung den Kanal/Instanz-Netzen verblüffend ähneln. Auch die materiellen Interpretationen der kreis- und rechteckförmigen graphischen Symbole KNÖLL's stimmen mit der analogen Bedeutung von Kanälen (Stellen) bzw. Instanzen (Transitionen) weitgehend überein.

2) Vgl. DITTRICH,G. (1989b), S. 2 ("Zu Beginn einer Systemmodellierung, also zur Beschreibung erster intuitiver Ideen, verwendet man Netze aus Kanälen und Instanzen"), S. 3 ("eignen ... sich besonders für erste intuitive Modellierungen") u. S. 8.

3) Oftmals werden Petrinetze schlechthin - nicht nur die spezielle Klasse der Kanal/Instanz-Netze - als reine Dokumentationsinstrumente verwendet.
Bei der reinen Dokumentationsfunktion von Netzen besteht kein Interesse an ihrer weiterführenden Analyse mit netztheoretischen Instrumenten. Entsprechend orientieren sich die Netzdarstellungen mehr an dem Kriterium ihrer intuitiven Verständlichkeit als an dem Kriterium ihrer formalen Ausdrucksmächtigkeit. Besonders beliebt sind in diesem reinen Dokumentationszusammenhang Stelle/Transition-Netze. Denn sie bleiben einerseits anschaulich, ohne daß die Einhaltung ihrer formalen Definition andererseits größere Schwierigkeiten bereiten würde. Bei dieser Verwendung des Petrinetz-Konzepts als ein reines Dokumentationskonzept geht jedoch das formale Ausdrucks- und Analysepotential des Petrinetz-Konzepts verloren. Daher folgt der Verf. diesem reduktionistischen Ansatz nicht. Statt dessen verwendet er zu reinen Dokumentationszwecken ausschließlich die besonders anschauliche Klasse der Kanal/Instanz-Netze.
Kanal/Instanz-Netze besitzen den Vorzug, durch das Instrument der semiformalen Netzbeschriftung wesentlich flexibler und ausdrucksstärker als Stelle/Transition-Netze auszufallen. Sobald jedoch auch analytische Erkenntnisse über die jeweils modellierten Objekte angestrebt werden, setzt der Verf. ein Netzmodell in der Gestalt eines Synthetischen Netzes voraus. In einem solchen Synthetischen Netz werden alle Modellierungsaspekte, die in einem Kanal/ Instanz-Netz noch natürlichsprachlich umschrieben wurden oder sogar nur implizit enthalten waren, in formalsprachlicher Weise wiedergegeben. Aufgrund seiner vollständig formalisierten Ausdrucksweise besitzt ein Synthetisches Netz als Netzmodell den Vorzug, einer breiten Palette formaler Analysekonzepte zugänglich zu sein.

4) Die Möglichkeit, die Präsentation eines prädikatenlogischen Objektmodells in Klausel- oder Produktionsregelform zu überspringen, wurde bereits aufgezeigt und gerechtfertigt.

5) Vgl. DITTRICH,G. (1989b), S. 4.

6) Die Semiformalität von Kanal/Instanz-Netzen hebt IGEL (1989a), S. 175, hervor. Allerdings legt er sich nicht darauf fest, die Semiformalität inhaltlich zu konkretisieren. Der semiformale Charakter klingt ebenso an bei DITTRICH,G. (1989b), S. 2f.

7) Dies rechtfertigt nachträglich, bei der ursprünglichen Entfaltung des Petrinetz-Konzepts nicht mit der Definition von Petrinetzen zu beginnen, sondern von Allgemeinen Netzen auszugehen.

8) Vgl. REISIG (1983a), S. 310 ("Channels, agencies and also arcs ... may hold *any* inscriptions."; kursive Hervorhebung durch den Verf.); IGEL (1986b), S. 3; DITTRICH,G. (1989b), S. 2.
In dieser Arbeit werden nur natürlichsprachliche Anschriften verwendet. Formalsprachliche Präzisierungen erfolgen hingegen beim Übergang zu Modellverfeinerungen, die auf Synthetischen Netzen basieren. Dies schließt jedoch keineswegs aus, in Kanal/Instanz-Netzen auch formalsprachliche Anschriften zu benutzen. Ein Beispiel mit umfangreichen formalsprachlichen Transitionsbeschriftungen findet sich z.B. bei WENDT (1989), S. 218, Bild 120.

9) Vgl. DITTRICH,G. (1989b), S. 3.

10) Diese Vorgehensweise klingt z.B. bei REISIG (1983a), S. 309 ("systematic development leading from informal to formal specifications"), an, wird dort aber nicht im Detail ausgeführt. Denn in den anschließenden Ausführungen bleibt es bei natürlichsprachlichen Netzanschriften.

11) Vgl. IGEL (1986b), S. 1 u. 3.

12) Daher wird für den semiformalen Charakter von Kanal/Instanz-Netzen auch der Grenzfall zugelassen, daß es sich um vollständig formalisierte Konstrukte handelt. Dieser Grenzfall wird in der Modellierungspraxis allerdings nur selten verwandt. Aus Gründen der Modellierungsökonomie wird statt dessen das vollständig formalisierte Kanal/Instanz-Netz nicht mehr explizit angegeben. An dessen Stelle tritt dann ein formalsprachliches prädikatenlogisches Objektmodell oder direkt ein entsprechendes Netzmodell. Dennoch bildet das vollständig formalisierte Kanal/Instanz-Netz immerhin noch ein implizites Zwischenglied in der Kette fortschreitender Formalisierung der Objektabbildung.
Selbst wenn ein vollständig formalisiertes Kanal/Instanz-Netz explizit vorgelegt wird, so fällt es noch keineswegs mit einem Netzmodell zusammen. Denn die Kanal/Instanz-Netze besitzen keine wohldefinierte dynamische Struktur. Für sie sind weder Schaltregeln noch Erreichbarkeitsgraphen vorgesehen. Netzmodelle verfügen dagegen als Stelle/Transition- oder Synthetische Netze über diese Konstituenten.

13) Auf den Präzisionsvorteil formaler Konzepte wurde bereits eingegangen. Der Aspekt der Netzverfeinerung durch zunehmende Formalisierung der Netzdarstellung kommt in Kürze zur Sprache.

14) Vgl. zur Charakterisierung der Stellen eines Kanal/Instanz-Netzes als passive Komponenten z.B. IGEL (1989a), S. 175; DITTRICH,G. (1989b), S. 3; SCHEER (1991d), S. 131.

15) Wenn die Objekte als Informationsdarstellungen aufgefaßt werden, entsprechen die passiven Komponenten den Kanälen aus dem Normenblatt DIN 66200; vgl. IGEL (1989a), S. 175.

16) Der Speichercharakter wird von WENDT (1989), S. 217ff., besonders hervorgehoben. Vgl. daneben auch DITTRICH,G. (1989b), S. 3.

17) Vgl. zur zustandsbezogenen Interpretation der Stellen eines Kanal/Instanz-Netzes DITTRICH,G. (1989b), S. 3; SCHEER (1991d), S. 131f.

18) Beispielsweise gibt in der nachfolgenden Abb. 56 die Stelle, die mit dem Ausdruck "wartende Aufträge" beschriftet ist, zunächst den zustandsorientierten Blickwinkel wieder: Sie repräsentiert die sachliche Bedingung, daß mindestens ein Auftrag in das Fertigungssystem eingelastet sein muß, damit eine Entscheidung gefällt werden kann, welcher Bearbeitungsstation dieser Auftrag zugeordnet wird. Ebenso läßt sich diese Stelle aber auch als eine passive Komponente auffassen, welche die eingelasteten, aber noch nicht zugeordneten Aufträge zwischenspeichert, wie z.B. ein Pufferlager. Passive und zustandsartige Perspektive lassen sich aber nicht immer gegenseitig austauschen. Wenn etwa im voranstehend behandelten Beispiel kein Pufferlager existiert, so repräsentiert die Stelle "wartende Aufträge" nur einen abstrakten Zustand, nicht aber eine reale passive Komponente des Fertigungssystems. Die jeweils betroffenen Aufträge sind dann zwar durch globale Einlastungsentscheidungen schon für die Produktion freigegeben worden, befinden sich aber real immer noch im Eingangslager des Fertigungssystems.

19) Vgl. zur Charakterisierung der Transitionen eines Kanal/Instanz-Netzes als aktive Komponenten z.B. IGEL (1989a), S. 175; DITTRICH,G. (1989b), S. 3; SCHEER (1991d), S. 131.

20) Dies entspricht der Instanzendefinition aus dem Normblatt DIN 66200, wenn Operationen und Tätigkeiten synonym behandelt werden. Vgl. IGEL (1989a), S. 175.

21) Im Beispiel der nachfolgenden Abb. 56 wurde die Transition, die mit dem Ausdruck "Zuordnungsentscheidungen" beschriftet sind, zunächst aus der vorgangsorientierten Perspektive konzeptualisiert. Sie repräsentiert die Entscheidungsoperation, durch deren Ausführungen Aufträge, Arbeitskräfte, Werkzeuge, Transportmittel und Bearbeitungstationen jeweils zu einer produktiven Einheit kombiniert werden. Ebenso hätte aber auch der Entscheidungsträger betrachtet werden können, so daß die gleiche Transition etwa mit dem Ausdruck "Arbeitsvorbereitung" oder "dispatch" beschriftet worden wäre.

22) Dabei kann es sich z.B. um Fertigstellungsmeldungen handeln, wenn die Bearbeitungsoperationen an einem Auftrag in einem einzelnen Bearbeitungsstation oder in einem ganzen Fertigungssystem abgeschlossen sind.

23) Aktorunabhängige Ereignisgeschehnisse sind z.B. Maschinenstörungen.

24) Die alternative Einteilungsmöglichkeit in zustands- und vorgangsartige Komponenten wird hier der Einfachheit halber nicht weiter erwähnt. Für sie gelten aber die nachfolgenden Ausführungen analog.

25) Diese Verfeinerung wird später durch die komplementären Instrumente der Makroknoten und Subnetze formal präzisiert und in das Konzept der Synthetische Netze eingebettet.

26) Sobald ein Netzknoten verfeinert wird, muß er in mindestens zwei Knoten aufgespalten werden. Es wird vorausgesetzt, daß alle verfeinerten Knoten ein zusammenhängendes Subnetz bilden. Andernfalls widerspräche es der Intuition des Verfeinerungsgedankens, daß eine Komponente eines Objektmodells in zusammenhangslose Subkomponenten zerlegt würde. Ein zusammenhängendes Netz mit mindestens zwei Knoten muß aber aus mindestens einer Stelle und mindestens einer Transition bestehen. Denn durch die zusammenhangstiftende Flußrelation können jeweils nur zwei *verschiedenartige* Knoten miteinander verbunden werden.

27) In dem Beispiel der nachfolgenden Abb. 56 werden etwa die "Bearbeitungsstationen" zunächst als passive Komponenten konzeptualisiert. Dort befinden sich die Aufträge, nachdem ihre Einlastung entschieden und bevor ihre Fertigstellung gemeldet wurde. In der späteren Fallstudie werden dagegen Bearbeitungsstationen weitaus feiner dargestellt. Dabei lassen sich Bearbeitungsoperationen identifizieren, die in den Bearbeitungsstationen ausgeführt werden. Sie werden im stationsrepräsentierenden Netzmodul durch entsprechende Transitionen repräsentiert. Umgekehrt kann aber auch eine Operation, die zunächst als eine atomare Operation konzeptualisiert wurde, zu einem Prozeß verfeinert werden, der aus mehreren Operationen zusammengesetzt ist. Dies trifft z.B. auf die Arbeitsgänge zu, die als atomare Operationen eingeführt worden waren. Sie werden in der Fallstudie als Prozesse behandelt, die aus einer Rüst- und einer Bearbeitungsoperation zusammengesetzt sind. Eine analoge Verfeinerung von zunächst atomaren Arbeitsgängen durch Prozesse, die aus Operationen zusammengesetzt sind, erwähnt auch HELBERG (1987), S. 184, im Zusammenhang mit der Zellensteuerung Flexibler Fertigungssysteme.

28) Vgl. zur Interpretationsvielfalt der Kanten eines Kanal/Instanz-Netzes DITTRICH,G. (1989b), S. 3.

29) Sie werden synonym als Anschriften oder Netzanschriften bezeichnet.

30) Dies ist jedoch keineswegs notwendig. Oftmals wird in Beiträgen zu Kanal/Instanz-Netzen die Möglichkeit ihrer Markierung sogar überhaupt nicht erwähnt oder sogar ausgeschlossen; vgl. z.B. IGEL (1986b), S. 2ff.; SCHEER (1991d), S. 131f.

31) Dies ist z.B. bei IGEL (1986b), S. 2, der Fall.

32) Z.B. werden die Objekte "Arbeitskräfte", "Werkzeuge" und "Transportmittel" auf diese Weise als Kantenanschriften angesprochen.

33) Falls Konstituentengruppen gebildet werden, die jeweils entweder nur aus Stellen oder nur aus Transitionen bestehen, so entsprechen diese Gruppen den "Klassenknoten" von IGEL (1986b), S. 16ff.

34) Im u.a. Beispiel werden etwa zwei Stellen mit der gemeinsamen Anschrift "Eingangspuffer" beschriftet, anstatt jede Stelle mit derselben Anschrift zu versehen.

35) Aus der Konzeption von Petrinetzen folgt, daß die Ein- und Ausgangskanten ihrer Netzknoten jeweils entweder disjunktiv oder aber konjunktiv verknüpft sind je nachdem, ob es sich um einen stellen- bzw. einen transitionsartigen Knoten handelt. "Besondere" logische Verknüpfungen liegen immer dann vor, wenn von dieser regulären Verknüpfung abgewichen wird. Auf die logischen Kantenverknüpfungen von Petrinetzen wird noch ausführlicher zurückgekommen.

36) Die *abgerundeten* Ecken unterscheiden die kastenförmigen Graphiksymbole für stellenartige Netzknoten eindeutig von den *recht*eckigen Graphiksymbolen für transitionsartige Netzknoten.

37) In dem Beispiel der nachfolgenden Abb. 56 werden Aufträge durch die Beschriftung einer Stelle mit dem Ausdruck "wartende Aufträge" ausgewiesen. Kantenanschriften identifizieren "Arbeitskräfte", "Werkzeuge" und "Transportmittel". Die gemeinsame Beschriftung "Bearbeitungsstationen" für mehrere Transitionen kennzeichnet die Betriebsmittel und zugehörigen Hilfseinrichtungen. Eine detaillierte formalsprachliche Transitionsbeschriftung präsentiert auch WENDT (1989), S. 218. Sie erklärt die Operationsweise eines Oszillators.

38) Vgl. IGEL (1986b), S. 1.

39) Vgl. IGEL (1986b), S. 3.

40) Der Schaltregelverzicht klingt mittelbar an bei IGEL (1989a), S. 175: Er betont, daß ein Kanal/Instanz-Netz unabhängig vom konkreten Schaltverhalten eines anderen, vollständig formalisierten Petrinetzes eine informale (treffender: semiformale) Beschreibung des gleichen Modellierungsobjekts leiste.

41) Hier wird besonders deutlich, wie treffend die Bezeichnung "Flußrelation" für die Komponente "F" aus der Definition Allgemeiner Netze durch das Tupel AN=(S,T;F) gewählt wurde.

42) Vgl. IGEL (1986b), S. 1f. u. 7ff.; DITTRICH,G. (1989b), S. 2 u. 9ff.

43) Dies bedeutet u.a., daß die Flußrelation des ursprünglichen Netzes als Teilmenge der Flußrelation des verfeinerten Netzes erhalten bleibt. Des weiteren wird dafür Sorge getragen, daß ein verfeinerter Netzknoten immer wieder ein Subnetz vom Kanal/Instanz-Typ liefert. Dadurch wird vor allem ausgeschlossen, daß bei der Knotenverfeinerung nur gleichartige Knoten erzeugt oder gleichartige Knoten durch kanten miteinander verknüpft werden. Die konsistenzwahrenden Verfeinerungsbedingungen werden von IGEL (1986b), S. 3ff., beschrieben und sind dort auf S. 5 präzise dokumentiert.

44) Besonders intensiv haben sich mit solchen Schichtenmodellen WEDEKIND und ZÖRNTLEIN auseinandergesetzt; vgl. WEDEKIND (1986a), S. 100ff., insbesondere S. 102ff.; WEDEKIND (1986b), S. 12ff.; WEDEKIND (1987a), S. 85ff.; WEDEKIND (o.J.a), S. 5ff.

45) Vgl. REFA (1985e), S. 82ff.

46) Es wird hier die Konzeptualisierung Flexibler Fertigungssysteme vorweggenommen, die später ausführlicher entfaltet wird. Daher wird die spezielle Konzeptualisierungsweise hier noch nicht gerechtfertigt, sondern kommentarlos präsentiert. Ein ähnliche Konzeptualisierung findet sich bei ALTING (1989a), S. 140ff., insbesondere Fig. 3 auf S. 141 (dort schrumpft das Flexible Fertigungssystem allerdings auf eine Flexible Fertigungszelle zusammen).

47) Ein Maßstab für die Schwierigkeit des Konzeptualisierungsprozesses i.w.S. wird in dieser Arbeit nicht vorgelegt. Statt dessen erfüllen die Ausführungen zur Konstruktion von Netzmodellen eine demarkative Funktion: Alle Netzmodelle, die im Anschluß an die natürlichsprachliche Objektmodellierung unmittelbar präsentiert werden, kennzeichnen den jeweils zugrundeliegenden Konzeptualisierungsprozeß i.w.S. als trivial oder leicht rekonstruierbar. Alle schrittweisen Herleitungen von Netzmodellen zeichnen dagegen die zugrundeliegenden Konzeptualisierungsprozesse i.w.S. als nicht-trivial. Dabei gelten jene Konzeptualisierungsprozesse als um so schwieriger, je mehr Herleitungsschritte offengelegt werden.

48) Diese Netzklasse wird stets vorausgesetzt, sofern bei der Veranschaulichung eines Sachverhalts durch den visualisierten Graphen eines Netzes nicht ausdrücklich darauf hingewiesen wird, daß das Netz zu einer anderen Netzklasse gehört. Dies gilt insbesondere auch für die "Netze", die zu Beginn dieser Arbeit benutzt wurden. Da die entsprechenden konzeptionellen und terminologischen Voraussetzungen noch nicht zur Verfügung standen, konnten diese "Netze" damals nicht korrekt als visualisierte Graphen von Netzen ausgewiesen werden, die zur Klasse der Kanal/Instanz-Netze gehören.

49) Auch die Kanal/Instanz-Netze, die jeweils ein natürlich- oder formalsprachliches Objektmodell veranschaulichen, das noch kein Synthetisches Netz darstellt, könnten als Netzmodelle (i.w.S.) bezeichnet werden. Um die Diktion trennschärfer zu gestalten, spricht der Verf. in allen diesen Fällen nur von Veranschaulichungen der Objektmodelle, während er den Begriff des Netzmodells für den eng gefaßten Anwendungsbereich von Synthetischen Netzen reserviert.

Literaturverzeichnis zu Teilband 5.1

Vorbemerkungen:

- Jedes Werk wird durch die Angabe eines Referenztitels (1. Zeile) und durch seine bibliographischen Angaben (folgende Zeilen) aufgeführt. In den Quellenangaben dieser Arbeit wird immer auf den Referenztitel Bezug genommen.

- Die Referenztitel bestehen nur aus den Autorennachnamen und den Erscheinungsjahren, solange hierdurch eine eindeutige Identifizierung der jeweils zugehörigen Werke möglich ist. Andernfalls dienen zusätzliche - abgekürzte - Autorenvornamen oder alphabetische Zusätze zu den Erscheinungsjahren der eindeutigen Identifizierung.

- Um eine einheitliche Quellenangabe in allen Bänden des Projekts PEMOPS zu gewährleisten, bezieht sich die eindeutige Identifizierung durch Autorenvornamen und alphabetische Zusätze zu den Erscheinungsjahren auf den Gesamtkorpus aller verarbeiteten Quellen. Daher kann es dazu kommen, daß innerhalb eines Bandes Lücken klaffen. Sie resultieren daraus, daß die scheinbar fehlenden Quellen im Gesamtkorpus zwar enthalten sind, aber im jeweils betroffenen Band nicht verwendet wurden.

- Die Titel fremdsprachlicher Werke werden grundsätzlich in der Notation des Originals wiedergegeben. Allerdings gelten drei Ausnahmen:
  - Titel, die sich nicht mit dem deutschsprachigen Alphabet ausdrücken lassen, werden in ihrer lautsprachlichen Umschreibung durch das deutschsprachige Alphabet wiedergegeben. Dies gilt insbesondere für Werke mit chinesischen oder kyrillischen Schriftzeichen.
  - Falls die Titel im Original durchgängig mit Großbuchstaben dargestellt werden, erfolgt hier eine Notation in der jeweils sprachspezifischen Groß-/Kleinschreibung von Titeln. Dies trifft vor allem auf anglophone Werke zu, in deren Titeln die jeweils sinnbestimmenden Worte durch Großbuchstaben eingeleitet werden.
  - Accents und andere diakritische Zeichenbestandteile, die nicht im deutschsprachigen Alphabet enthalten sind, werden grundsätzlich ausgelassen.

- In das Literaturverzeichnis wurden alle Quellen aufgenommen, auf die in den Anmerkungen zum laufenden Text verwiesen wurde.

- Weitere Publikationen, die sich auf die Thematik des Petrinetz-Konzepts beziehen, aber in den vorgenannten Quellen nicht angesprochen wurden, finden sich im Band 10 des Projekts PEMOPS zur Petrinetz-Literatur.

- Die Literaturauswertung wurde 1992 abgeschlossen (vgl. das Vorwort in Band 1).

Abel,D. (1990)
Abel,D.: Petri-Netze für Ingenieure - Modellbildung und Analyse diskret gesteuerter Systeme, Berlin - Heidelberg - New York 1990.

Alting (1989a)
Alting,L.; Christensen,S.C.; Pedersen,M.A.: An Integrated Miniature Laboratory for Research and Education; in: Kochan,D.; Olling,G. (Hrsg.): Software for Manufacturing, Proceedings of the Seventh International IFIP/IFAC Conference on Software for Computer Integrated Manufacturing, PROLAMAT'88, 14.-17.06.1988 in Dresden, Amsterdam - New York - Oxford ... 1989, S. 135-144.

Appelrath (1983)
Appelrath,H.-J.: Konzepte der Wissensbereitstellung in Expertensystemen: Inferenzmechanismen auf relationalen Datenbanken, Dissertation, Universität Dortmund, Dortmund 1983.

Bancilhon (1986)
Bancilhon,F.; Khoshafian,S.: A Calculus for Complex Objects; in: o.V.: Proceedings of the Fifth ACM SIGACT-SIGMOD Symposium on Principles of Database Systems, 24.-26.03.1986 in Cambridge (Massachusetts), New York 1986, S- 53-59.

Barr,R. (1989)
Barr,R.S.; Christiansen,M.G.: A Parallel Auction Algorithm: A Case Study in the Use of Parallel Object-Oriented Programming; in: Sharda,R.; Golden,B.L.; Wasil,E.; Balci,O.; Stewart,W. (Hrsg.): Impacts of Recent Computer Advances on Operations Research, New York - Amsterdam - London 1989, S. 23-32.

Battiston (1988)
Battiston,E.; De Cindio,F.; Mauri,G.: OBJSA Nets: a class of high-level nets having objects as domains; in: Rozenberg,G. (Hrsg.): Advances in Petri Nets 1988, Lecture Notes in Computer Science 340, Berlin - Heidelberg - New York ... 1988, S. 20-43.

Bauman (1986)
Bauman,R.; Turano,T.A.: Production based language simulation of Petri nets; in: Simulation, Vol. 47 (1986), S. 191-198.

Bibel (1989)
Bibel,W.; del Cerro,L.F.; Fronhöfer,B.; Herzig,A.: Plan Generation by Linear Proofs: On Semantics; in: Metzing,D. (Hrsg.): GWAI-89, 13th German Workshop on Artificial Intelligence, 18.-22.09.1989 in Eringerfeld, Proceedings, Informatik-Fachberichte 216, Berlin - Heidelberg - New York ... 1989, S. 49-62.

Billington (1982)
Billington,J.: Specification of the Transport Service Using Numerical Petri Nets; in: Sunshine,C. (Hrsg.): Protocol Specification, Testing, and Verification II, Proceedings of the IFIP WG 6.1 Second International Workshop on Protocol Specification, Testing, and Verification, 1982 in Idyllwild, Amsterdam - Oxford - New York 1982, S. 77-100.

Billington (1983)
Billington,J.: Abstract Specification of the ISO Transport Service Definition Using Labelled Numerical Petri Nets; in: Rudin,H.; West,C.H. (Hrsg.): Protocol Specification, Testing, and Verification, III, Proceedings of the IFIP WG 6.1 Third International Workshop on Protocol Specification, Testing, and Verification, 31.05.-02.06.1983 in Rüschlikon, Amsterdam - New York - Oxford 1983, S. 173-185.

Böhringer (1988)
Böhringer,B.; Chiopris,C.; Futo,I.: Wissensbasierte Systeme mit Prolog, Bonn - Reading - Massachusetts ... 1988.

Bossi (1989)
Bossi,A.; Cocco N.: Verifying correctness of logic programs; in: Diaz,J.; Orejas,F. (Hrsg.): TAPSOFT'89, Proceedings of the International Joint Conference on Theory and Practice of Software Development, 13.-17.03.1989 in Barcelona, Volume 2: Advanced Seminar on Foundations of Innovative Software Development II and Colloquium on Current Issues in Programming Languages (CCIPL), Lecture Notes in Computer Science 352, Berlin - Heidelberg - New York ... 1989, S. 96-110.

Bucher (1987)
Bucher,T.G.: Einführung in die angewandte Logik, Berlin - New York 1987.

Bunge (1977)
Bunge,M.: Treatise on Basic Philosophy, Volume 3, Ontology I: The Furniture of the World, Dordrecht - Boston 1977.

Chen,P.P. (1976)
Chen,P.P.-S.: The Entity-Relationship Model - Toward a Unified View of Data; in: ACM Transactions on Database Systems, Vol. 1 (1976), S. 9-36.

Clocksin (1990)
Clocksin,W.F.; Mellish,C.S.: Programmieren in Prolog, (Übersetzung der 3. Aufl. 1987), Berlin - Heidelberg - New York ... 1990.

Codd (1970)
Codd,E.F.: A Relational Model of Data for Large Shared Data Banks; in: Communications of the ACM, Vol. 13 (1970), S. 377-387.

Codd (1971a)
Codd,E.F.: Normalized Data Base Structure: A Brief Tutorial; in: Codd,E.F.; Dean,A.L. (Hrsg.): Proceedings of the 1971 ACM-SIGFIDET Workshop on Data Description, Access and Control, 11.-12.11.1971 in San Diego, New York 1971, S. 1-17.

Codd (1972)
Codd,E.F.: Relational Completeness of Data Base Sublanguages; in: Rustin,R. (Hrsg.): Data Base Systems, Englewood Cliffs 1972, S. 65-98.

Commission of the European Communities (1989)
Commission of the European Communities: 1989 ESPRIT Workprogramme (European Strategic Programme for R&D in Information Technology), Brüssel 1989.

Cordes (1988)
Cordes,R.; Kruse,R.; Langendörfer,H.; Rust,H.: Prolog - Eine methodische Einführung, Braunschweig - Wiesbaden 1988.

Davis,R. (1975)
Davis,R.; King,J.: An Overview of Production Systems, Memo AIM-271, Stanford Artificial Intelligence Laborytory, zugleich: Report No. STAN-CS-75-524, Computer Science Department, University of Stanford, Stanford 1975.

de Saussure (1967)
de Saussure,F.: Grundfragen der allgemeinen Sprachwissenschaft, 2. Aufl., Berlin 1967.

Dittrich,G. (1989b)
Dittrich,G.; Evertz-Jägers,B.: Der Kanal-Instanz-Netz Editor KINED - Ein Tool zur Unterstützung einer methodischen Systemmodellierung mit Hilfe von hierarchisch dargestellten Kanal-Instanz-Netzen, Forschungsbericht Nr. 308, Fachbereich Informatik, Universität Dortmund, Dortmund o.J. (1989).

Dittrich,K. (1985)
Dittrich,K.R.; Kotz,A.M.; Mülle,J.A.; Lockemann,P.C.: Datenbankunterstützung für den ingenieurwissenschaftlichen Entwurf - Eine Übersicht über den Stand der Entwicklung; in: Informatik-Spektrum, Bd. 8 (1985), S. 113-125.

Dittrich,K. (1989)
Dittrich,K.R.: Objektorientierte Datenbanksysteme; in: Informatik-Spektrum, Bd. 12 (1989), S. 215-220.

Dorn (1989)
Dorn,J.: Wissensbasierte Echtzeitplanung, Dissertation, Technische Universität Berlin, Braunschweig - Wiesbaden 1989.

Drosten (1989)
Drosten,K.: Termersetzungssysteme - Grundlagen der Prototyp-Generierung algebraischer Spezifikationen, überarbeitete Dissertation, Technische Universität Braunschweig 1987, Informatik-Fachberichte 210, Berlin - Heidelberg - New York ... 1989.

Eschenbacher (1989)
Eschenbacher,P.: Die Modellbeschreibungssprache SIMPLEX-MDL; in: Pressmar,D.; Jäger,K. E.; Krallmann,H.; Schellhaas,H.; Streitferdt,L. (Hrsg.): Operations Research Proceedings 1988, DGOR - Vorträge der 17. Jahrestagung, 13.-16.09.1988 in Berlin, Berlin - Heidelberg - New York ... 1989, S. 119-125.

Eschenbacher (1991)
Eschenbacher,P.: Formulierung transactions-orientierter Modelle mit der systemtheoretischen Beschreibungssprache SIMPLEX-MDL; in: Biethahn,J.; Hummeltenberg,W.; Schmidt,B. (Hrsg.): Simulation als betriebliche Entscheidungshilfe, Band 2, Berlin - Heidelberg - New York ... 1991, S. 221-235.

Esser,H. (1977a)
Esser,H.; Klenovits,K.; Zehnpfennig: Wissenschaftstheorie 1: Grundlagen und Analytische Wissenschaftstheorie, Stuttgart 1977.

Fehling (1990a)
Fehling,R.: Hierarchische Petrinetze - Idee und grundlegende Struktur, Forschungsbericht Nr. 344, Fachbereich Informatik, Universität Dortmund, Dortmund 1990.

Fidelak (1986a)
Fidelak,M.: Wissensdarstellung und -verarbeitung auf der Basis von Petri-Netzen, Diplomarbeit am Fachbereich Informatik/Universität Bonn, Bonn 1986. (Auch veröffentlicht als: Arbeitspapiere der GMD Nr. 225, Gesellschaft für Mathematik und Datenverarbeitung mbH/Bonn, Sankt Augustin 1986.)

Fidelak (1986b)
Fidelak,M.: Petri-Netze - Eine formale Sprache zur Wissensrepräsentation; in: Rundbrief des Fachausschusses 1.2 Künstliche Intelligenz & Mustererkennung in der Gesellschaft für Informatik, Nr. 43 (1986), S. 32-38.

Fidelak (1987a)
Fidelak,M.: Petri Nets for Logical Representation; in: Chorafas,D.N.; Rowe,A.J. (Hrsg.): 1st International Symposium on Artificial Intelligence and Expert Systems, 18.-22.05.1987 in Berlin, Proceedings, Part A, Berlin 1987, S. 95-116.

Freedman (1988b)
Freedman,P.; Malowany,A.: SAGE: A Decision Support System for the Sequencing of Operations within a Robotic Workcell; in: Decision Support Systems, Vol. 4 (1988), S. 329-343.

Friedrichs,J. (1987)
Friedrichs,J.: Expertensystemansatz zum Entwurf verteilter Systeme, Mitteilung DFVLR-Mitt. 87-16, Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt, zugleich Diplom-Arbeit, Institut für Flugmechanik, Technische Universität Braunschweig, Köln 1987.

Fronhöfer (1987)
Fronhöfer,B.: PLANLOG: A Language Framework for the Integration of Procedural and Logical Programming; in: o.V.: IJCAI 87, Proceedings of the Tenth International Joint Conference on Artificial Intelligence, 23.-28.08.1987 in Mailand, o.O. (Los Altos) 1987, Vol. 1, S. 15-17.

Genrich (1987a)
Genrich,H.J.: Predicate/Transition Nets; in: Brauer,W.; Reisig,W.; Rozenberg,G. (Hrsg.): Petri Nets: Applications and Relationships to Other Models of Concurrency, Advances in Petri Nets 1986, Part I, Proceedings of an Advanced Course, 8.-19.09.1986 in Bad Honnef, Lecture Notes in Computer Science 254, Berlin - Heidelberg - New York ... 1987, S. 207-247.

Genrich (1988b)
Genrich,H.J.: Equivalence Transformations of PrT-Nets; in: o.V.: Application and Theory of Petri Nets - 9th European Workshop, 22.-24.06.1988 in Venedig, o.O. 1988, Vol. I, S. 229-248.

Genrich (1990b)
Genrich,H.: High Level Nets, Invited Talk, gehalten am 27.06.1990 in Paris anläßlich: 11th International Conference on Application and Theory of Petri Nets, 27.-29.06.1990 in Paris.

Georgeff (1986)
Georgeff,M.P.: The Representation of Events in Multiagent Domains; in: o.V.: Proceedings AAAI-86, Fifth International Conference on Artificial Intelligence, 11.-15.08.1986 in Philadelphia, Los Altos 1986, Vol. 1, S. 70-75.

Haas (1987)
Haas,P.J.; Shedler,G.S.: Stochastic Petri Nets with Simultaneous Transition Firings; in: o.V.: International Workshop on Petri Nets and Performance Models, PNPM87, 24.-26.08.1987 in Madison, Washington 1987, S. 24-32.

Helberg (1987)
Helberg,P.: PPS als CIM-Baustein - Gestaltung der Produktionsplanung und -steuerung für die computerintegrierte Produktion, Berlin 1987.

Hinst (1983)
Hinst,P.: Quines Ontologiekriterium; in: Erkenntnis, Vol. 19 (1983), S. 193-215.

Hohenstein (1986)
Hohenstein,U.; Neugebauer,L.; Saake,G.: An Extended Entity-Relationship Model for Non-Standard Datenbases; in: Heuer,A. (Hrsg.): Workshop über Relationale Datenbanken, 16.-20.06.1986 in Lessach, Informatik-Bericht 86/3, Institut für Informatik, Universität Clausthal-Zellerfeld, Clausthal-Zellerfeld 1986, S. 185-211. (Anmk. des Verf.: "Datenbases" im Original; gemeint ist wohl: "Databases".)

Hohenstein (1987)
Hohenstein,U.; Neugebauer,L.; Saake,G.; Ehrich,H.-D.: Three-Level-Specification of Databases Using an Extended Entity-Relationship Model; in: Wagner,R.R.; Traunmüller,R.; Mayr,H.C. (Hrsg.): Informationsbedarfsermittlung und -analyse für den Entwurf von Informationssystemen, Fachtagung EMISA, 02.-03.07.1987 in Linz, Proceedings, Informatik-Fachberichte 143, Berlin - Heidelberg - New York ... 1987, S. 58-88.

Holt,A. (1975d)
Holt,A.W.: Communication Mechanics; in: Massachusetts Computer Associates, Inc. (Hrsg.): Second Semi-Annual Technical Report (1.12.1973-31.03.1975) for the Project "Development of Theoretical Foundations for Description and Analysis of Discrete Information Systems", Wakefield 1975, S. 156-176.

Holzmann (1987)
Holzmann,G.J.: On Limits and Possibilities of Automated Protocol Analysis; in: Rudin,H.; West,C.H. (Hrsg.): Protocol Specification, Testing, and Verification, VII, Proceedings of the IFIP WG 6.1 International Workshop on Protocol Specification, Testing, and Verifivation, 05.-08.05.1987 in Zürich, Amsterdam 1987, S. 339-344.

Hsiang (1983a)
Hsiang,J.; Dershowitz,N.: Rewrite Methods for Clausal and Non-Clausal Theorem Proving; in: Diaz,J. (Hrsg.): Automata, Languages and Programming, 10th Colloquium, 18.-22.07.1983 in Barcelona, Lecture Notes in Computer Science 154, Berlin - Heidelberg - New York .... 1983, S. 331-346.

Igel (1986b)
Igel,B.; Jungfermann,M.: Rechnergestützte graphische Spezifikation mit Kanal/Instanz-Netzen, Forschungsbericht Nr. 223, Abteilung Informatik, Universität Dortmund, Dortmund 1986.

Igel (1989a)
Igel,B.: Applikative Beschreibungsmethoden in verteilten Prozeßsteuerungen; in: Henn,R.; Stieger,K. (Hrsg.): PEARL 89 - Workshop über Realzeitsysteme, 10. Fachtagung des PEARL-Vereins e.V., 07.-08.12.1989 in Boppard, Proceedings, Informatik-Fachberichte 231, Berlin - Heidelberg - New York ... 1989, S. 172-195.

Itter (1989g)
Itter,F.; Relewicz,C.: Computer Supported Design of Kanban Controlled Production Integrated System Analysis and Simulation, Interner Bericht, PSI GmbH (Gesellschaft für Prozeßsteuerungs- und Informationssysteme mbH), o.O. (Berlin) o.J. (1989).

Jackson,M.A. (1979)
Jackson,M.A.: Grundsätze des Programmentwurfs, Darmstadt 1979.

Jensen (1987a)
Jensen,K.: Coloured Petri Nets; in: Brauer,W.; Reisig,W.; Rozenberg,G. (Hrsg.): Petri Nets: Applications and Relationships to Other Models of Concurrency, Advances in Petri Nets 1986, Part I, Proceedings of an Advanced Course, 8.-19.09.1986 in Bad Honnef, Lecture Notes in Computer Science 254, Berlin - Heidelberg - New York ... 1987, S. 248-299.

Kinnebrock (1988)
Kinnebrock,W.: Turbo Prolog, München - Wien 1988.

Kleiner,F. (1991)
Kleiner,F.: Kostenrechnung bei flexibler Automatisierung, Dissertation, Universität Stuttgart, München 1991.

Knöll (1989)
Knöll,H.-D.; Suk,W.: Für kommerzielle Programme: Grafische Sprache; in: computer magazin, 19. Jg. (1989), Heft 9, S. 49-52.

Kochan,D. (1986)
Kochan,D. (Hrsg. u. Autor); Merchant,o.Vn.; Kozar,o.Vn.; Schaller,J.; Hutchinson,G.K.; Olling, o.Vn.; Semenkov,o.Vn.; Klimov,W.; Spur,G.; Krause,F.L.; Pistorius,E.; Crestin,J.P. (Koautoren): CAM Developments in Computer-Integrated Manufacturing, Berlin - Heidelberg - New York - Tokyo 1986.

Krämer (1987a)
Krämer,B.; Schmidt,H.W.: Types and Modules for Net Specifications; in: Voss,K.; Genrich,H.J.; Rozenberg,G. (Hrsg.): Concurrency and Nets - Advances in Petri Nets, Berlin - Heidelberg - New York ... 1987, S. 269-286.

Lautenbach (1985b)
Lautenbach,K.: On Logical and Linear Dependencies, Arbeitspapiere der GMD 147, Gesellschaft für Mathematik und Datenverarbeitung mbH/Bonn, Sankt Augustin 1985.

Lee,R. (1988a)
Lee,R.M.: Logic, Semantics and Data Base Modeling: An Ontology; in: Meersman,R.A.; Sernadas,A.C. (Hrsg.): Data and Knowledge (DS-2), Proceedings of the Second IFIP 2.6 Working Conference on Database Semantics, 'Data and Knowledge' (DS-2), 03.-07.11.1986 in Albufeira, Amsterdam - New York - Oxford ... 1988, S. 221-243.

Lindgreen (1987)
Lindgreen,P.: Entities from a Systems Point of View; in: Spaccapietra,S. (Hrsg.): Entity-Relationship Approach: Ten Years of Experience in Information Modeling, Proceedings of the Fifth International Conference on Entity-Relationship Approach, 17.-19.11.1986 in Dijon, Amsterdam - New York - Oxford ... 1987, S. 119-131.

Liu,N. (1987)
Liu,N.K.; Dillon,T.: Detection of consistency and completeness in expert systems using Numerical Petri Nets; in: Gero,J.S.; Stanton,R. (Hrsg.): Artificial Intelligence Developments and Applications, Edited Selection of Papers to the Australian Joint Artificial Intelligence Conference, 02.-04.11.1987 in Sydney, Amsterdam - New York - Oxford ... 1987, S. 119-134.

Lu,M. (1987)
Lu,M.; Zhang,D.; Murata,T.: Stochastic Net Model for Self-Stability Measures of Fault Tolerant Clock Synchronization; in: o.V.: International Workshop on Petri Nets and Performance Models, PNPM87, 24.-26.08.1987 in Madison, Washington 1987, S. 104-110.

Mainz (1984)
Mainz,U.: Netztheoretische Repräsentation prädikatenlogischer Begriffe und Methoden, Diplomarbeit am Institut für Informatik, Universität Bonn, Bonn 1984.

Marti-Oliet (1989)
Marti-Oliet,N.; Meseguer,J.: From Petri Nets to Linear Logic; in: Pitt,D.H.; Rydeheard,D.E.; Dybjer,P.; Pitts,A.M.; Poigne,A. (Hrsg.): Category Theory and Computer Science, 05.-08.09.1989 in Manchester, Proceedings, Lecture Notes in Computer Science 389, Berlin - Heidelberg - New York ... 1989, S. 313-340.

Mayr,H. (1987)
Mayr,H.C.; Dittrich,K.R.; Lockemann,P.C.: Datenbankentwurf; in: Lockemann,P.C.; Schmidt,J.W. (Hrsg.): Datenbank-Handbuch, Berlin - Heidelberg - New York ... 1987, S. 481-557.

Murata,Ta. (1988a)
Murata,Ta.; Matsuyama,K.: Inconsistency Check of a Set of Clauses using Petri Net Reductions; in: Journal of the Franklin Institute, Vol. 325 (1988), No. 1, S. 73-93.

Murata,Ta. (1988b)
Murata,Ta.; Zhang,D.: A Predicate-Transition Net Model for Parallel Interpretation of Logic Programs; in: IEEE Transactions on Software Engineering, Vol. SE-14 (1988), S. 481-497.

Nakano,R. (1983)
Nakano,R.: Integrity Checking in a Logic-Oriented ER Model; in: Davis,C.G.; Jajodia,S.; Ng,P.A.-B.; Yeh,R.T. (Hrsg.): Entity-Relationship Approach to Software Engineering, Proceedings of the Third International Conference on Entity-Relationship Approach, 05.-07.10.1983 in Anaheim, Amsterdam - New York - Oxford 1983, S. 551-564.

Neumann,T. (1983)
Neumann,T.: Konzepte zur Erweiterung von Datenbanksystemen für die Unterstützung von CAD/CAM-Anwendungen, Dissertation, Technische Hochschule Darmstadt, Darmstadt 1983.

Niehuis (1986)
Niehuis,S.; Victor,F.: Modellierung und Simulation von Pr/T-Netzen in Prolog, Arbeitspapiere der GMD 231, Gesellschaft für Mathematik und Datenverarbeitung mbH/Bonn, Sankt Augustin 1986.

Oberquelle (1987a)
Oberquelle,H.: Human-Machine Interaction and Role/Function/Action-Nets; in: Brauer,W.; Reisig,W.; Rozenberg,G. (Hrsg.): Petri Nets: Applications and Relationships to Other Models of Concurrency, Advances in Petri Nets 1986, Part II, Proceedings of an Advanced Course, 8.-19.09.1986 in Bad Honnef, Lecture Notes in Computer Science 255, Berlin - Heidelberg - New York ... 1987, S. 171-190.

Oberweis (1987b)
Oberweis,A.; Schönthaler,F.; Seib,J.; Lausen,G.: Database Supported Analysis Tool for Predicate/Transition Nets; in: Special Interest Group on Petri Nets and related System Models, Gesellschaft für Informatik, Newsletter 28 (1987), S. 21-23.

Oberweis (1988a)
Oberweis,A.; Seib,J.: PASIPP: Ein Programm zur Analyse und Simulation von Prolog-beschrifteten Prädikate/Transitionen-Netzen (Benutzerhandbuch), o.O. (Mannheim) 1988.

Oberweis (1988b)
Oberweis,A.: Checking Database Integrity Constraints while Simulating Information System Behaviour; in: o.V.: Application and Theory of Petri Nets - 9th European Workshop, 22.-24.06.1988 in Venedig, o.O. 1988, Vol. I, S. 299-308.

Oberweis (1988c)
Oberweis,A.; Seib,J.: PASIPP: Ein Programm zur Analyse und Simulation von Prolog-beschrifteten Prädikate/Transitionen-Netzen (Benutzerhandbuch zur Version 2.1), Fakultät für Mathematik und Informatik, Universität Mannheim, Mannheim 1988.

Oberweis (1989a)
Oberweis,A.; Seib,J.; Lausen,G.: Ein Programm zur Analyse und Simulation von Prädikate-Transitionen-Netzen, Interner Bericht, Fakultät für Mathematik und Informatik, Universität Mannheim, Mannheim 1989.

o.V. (1989g)
o.V.: Analyse und Simulation von verteilten Systemen, Prospekt, Fakultät für Mathematik und Informatik, Universität Mannheim, Mannheim o.J. (1989).

Pagnoni (1990)
Pagnoni,A.: Project Engineering - Computer-Oriented Planning and Operational Decision Making, Berlin - Heidelberg - New York ... 1990.

Petri,C. (1976a)
Petri,C.A.: Nicht-sequentielle Prozesse; in: Billing,H.; Händler,W.; Herzog,U.; Hofmann,F.; Leeb,K.; Mertens,P.; Müller,H.; Niemann,H.; Seitzer,D.; Schneider,H.J. (Hrsg.): Arbeitsberichte des Instituts für Mathematische Maschinen und Datenverarbeitung, Universität Erlangen-Nürnberg, Bd. 9, Nr. 8: Parallelismus in der Informatik, Wissenschaftliches Kolloquium aus Anlass des zehnjährigen Bestehens des IMMD Erlangen, 10.-11.06.1976, Erlangen 1976, S. 57-80. (Englische Übersetzung erschienen als: Petri (1977d).)

Petri,C. (1977d)
Petri,C.A.: Non-Sequential Processes, Report ISF-77-05, Institut für Informationssystemforschung, Gesellschaft für Mathematik und Datenverarbeitung mbH/Bonn, Sankt Augustin 1977. (Anmk. des Verf.: Übersetzung eines Vortrags über "Parallelism in Computer Science", gehalten im Juni 1976 an der Universität Erlangen-Nürnberg; zugleich ebenso Übersetung von Petri (1976a).)

Petri,C. (1979c)
Petri,C.A.: Über einige Anwendungen der Netztheorie; in: Böhling,K.H.; Spies,P.P. (Hrsg.): GI - 9. Jahrestagung, 01.-05.10.1979 in Bonn, Informatik-Fachberichte 19, Berlin - Heidelberg - New York 1979, S. 81-87.

Post,E. (1943)
Post,E.L.: Formal Reductions of the General Combinatorial Decision Problem; in: American Journal of Mathematics, Vol. 65 (1943), S. 197-215.

PROLOG (o.J.)
o.V.: Turbo Prolog Owner's Handbook, o.O. o.J.

Reck (1988)
Reck,M.: Von informellen zu formalen Spezifikationen durch Petri-Netze und abstrakte Datentypen, Diplomarbeit, Lehrstuhl 1, Fachbereich Informatik, Universität Dortmund, Dortmund 1988.

REFA (1985e)
REFA - Verband für Arbeitsstudien und Betriebsorganisation e.V. (Hrsg.): Methodenlehre der Planung und Steuerung, Teil 5, 4. Aufl., München 1985.

Reisig (1983a)
Reisig,W.: System Design Using Petri Nets; in: o.V.: Papers presented at the 4th European Workshop on Applications and Theory of Petri Nets, 26.-29.09.1983 in Toulouse, o.O. 1983, S. 309-321. (Auch erschienen in: Hommel,G.; Krönig,D. (Hrsg.): Requirements Engineering, Arbeitstagung der GI, 12.-14.10.1983 in Friedrichshafen, Informatik-Fachberichte 74, Berlin - Heidelberg - New York - Tokyo 1983, S. 29-41.)

Reisig (1985d)
Reisig,W.: Petri Nets with Individual Tokens; in: Theoretical Computer Science, Vol. 41 (1985), S. 185-213.

Reisig (1989a)
Reisig,W.: Petri Nets and Abstract Data Types, Bericht TUM-I8904, Institut für Informatik, Technische Universität München, München 1989.

Reisig (1989b)
Reisig,W.: Cover Picture Story - The Container Crane System; in: Petri Net Newsletter (Gesellschaft für Informatik: Special Interest Group on Petri Nets and related System Models), No. 34 (1989), S. 3-9.

Richter,G. (1983b)
Richter,G.: Netzmodelle für die Bürokommunikation - Teil 1; in: Informatik-Spektrum, Bd. 6 (1983), S. 210-220.

Richter,G. (1985b)
Richter,G.; Brücher,M.R.: Ein Netzmodell der referatsübergreifenden Vorgangsbearbeitung in der Ministerialverwaltung, Arbeitspapiere der GMD 150, Gesellschaft für Mathematik und Datenverarbeitung mbH/Bonn, Sankt Augustin 1985.

Richter,G. (1988b)
Richter,G.; Humphreys,P.; Genrich,H.; Berkeley,D.; Voss,K.; Domke,M.; Griebler,H.; Wisudha, A.: Generic Office Frame of Reference; in: Schäfer,G.; Hirschheim,R.; Harper,M.; Hansjee,R.; Domke,M.; Bojrn-Andersen,N. (Hrsg.): Functional Analysis of Office Requirements: A Multiperspective Approach, Chichester - New York - Brsibane ... 1988, S. 123-179.

Rosenstengel (1991)
Rosenstengel,B.; Winand,U.: Petri-Netze - Eine anwendungsorientierte Einführung, 4. Aufl., Braunschweig - Wiesbaden 1991.

Scheer (1987a)
Scheer,A.-W.: EDV-orientierte Betriebswirtschaftslehre, 3. Aufl, Berlin - Heidelberg - New York ... 1987.

Scheer (1988a)
Scheer,A.-W.: Wirtschaftsinformatik - Informationssysteme im Industriebetrieb, 2. Aufl., Berlin - Heidelberg - New York ... 1988.

Scheer (1989c)
Scheer,A.-W.: Enterprise-Wide Data Modelling - Information Systems in Industry, Berlin - Heidelberg - New York ... 1989.

Scheer (1991d)
Scheer,A.-W.: Architektur integrierter Informationssysteme - Grundlagen der Unternehmensmodellierung, Berlin - Heidelberg - New York ... 1991.

Schlageter (1983)
Schlageter,G.; Stucky,W.: Datenbanksysteme: Konzepte und Modelle, 2. Aufl., Stuttgart 1983.

Schmitz,P. (1985)
Schmitz,P.; Seibt,D.: Einführung in die anwendungsorientierte Informatik, Bd. 1: Systemtechnische Grundlagen, 3. Aufl., München 1985.

Sinzig (1983)
Sinzig,W.: Datenbankorientiertes Rechnungswesen - Grundzüge einer EDV-gestützten Realisierung der Einzelkosten- und Deckungsbeitragsrechnung, Berlin - Heidelberg - New York ... 1983.

Stahlknecht (1991)
Stahlknecht,P.; Appelfeller,W.; Drasdo,A.; Meier,H.; Nieland,S.: Arbeitsbuch Wirtschaftsinformatik, Berlin - Heidelberg - New York ... 1991.

Steinmetz,Ra. (1987a)
Steinmetz,Ra.: Relationship between Petri Nets and the Synchronisation Mechanisms of the Concurrent Languages Chill and Occam; in: o.V.: Papers presented at the 8th European Workshop on Application and Theory of Petri Nets, 24.-26.06.1987 in Zaragoza, o.O. 1987, S. 509-529.

Systems Modeling Corporation (o.J.b)
Systems Modeling Corporation (Hrsg.): Cinema - simulation animation ... like you've never seen before!, Sewickley o.J.

Tabourier (1983b)
Tabourier,Y.: Further Development of the Occurences Structure Concept: The EROS Approach; in: Davis,C.G.; Jajodia,S.; Ng,P.A.-B.; Yeh,R.T. (Hrsg.): Entity-Relationship Approach to Software Engineering, Proceedings of the Third International Conference on Entity-Relationship Approach, 05.-07.10.1983 in Anaheim, Amsterdam - New York - Oxford 1983, S. 565-583.

Tempelmeier,H. (1989a)
Tempelmeier,H.: Simulation mit SIMAN, 6. Aufl., Institut für Betriebswirtschaftslehre, Fachgebiet Fertigungs- und Materialwirtschaft, Technische Hochschule Darmstadt, Darmstadt 1989.

Thieler-Mevissen (1975)
Thieler-Mevissen,G.: Vollständigkeit und Korrektheit des netztheoretischen Kalküls für die Aussagenlogik, Interner Bericht 04/75-5-9, Gesellschaft für Mathematik und Datenverarbeitung mbH/Bonn, Sankt Augustin 1975.

Thieler-Mevissen (1977)
Thieler-Mevissen,G.: The Petri Net Calculus of Predicate Logic, Interner Bericht ISF-76-09, Institut für Informationssystemforschung, Gesellschaft für Mathematik und Datenverarbeitung mbH/Bonn, Sankt Augustin 1977.

Thieler-Mevissen (1987a)
Thieler-Mevissen,G.: Existential Quantifiers in Predicate-Fact-Nets; in: Voss,K.; Genrich,H.J.; Rozenberg,G. (Hrsg.): Concurrency and Nets - Advances in Petri Nets, Berlin - Heidelberg - New York ... 1987, S. 533-553.

Thieler-Mevissen (1987b)
Thieler-Mevissen,G.: Anwendungen der Netztheorie zur formalen Systemspezifikation, GMD-Bericht Nr. 166, München - Wien 1987.

Vagin (1988)
Vagin,V.N.; Zakharov,V.N.; Rozenblyum,L.Y.: Logical Inference on Interpreted Petri Nets; in: Soviet Journal of Computer and Systems Sciences, Vol. 26 (1988), No. 3, S. 98-105.

Varney (1988)
Varney,L.R.; Swiderski,D.: Real-Time Knowledge-Based Programming with IF/tasklog: An Executive for Asynchronous Multitasking in IF/Prolog, IF/PROmLOG, and C, Paper, Interface Computer GmbH, München o.J. (1988).

Vautherin (1987b)
Vautherin,J.: Parallel Systems Specifications with Coloured Petri Nets and Algebraic Specifications.; in: Rozenberg,G. (Hrsg.): Advances in Petri Nets 1987, Lecture Notes in Computer Science 266, Berlin - Heidelberg - New York ... 1987, S. 293-308.

von Luck (1989)
v. Luck,K.; Meyer,R.; Pirlein,T.: Die logische Rekonstruktion eines Gegenstandbereiches - Eine Fallstudie -; in: Retti,J.; Leidlmair,K. (Hrsg.): 5. Österreichische Artificial-Intelligence-Tagung, 28.-31.03.1989 in Igls, Proceedings, Informatik-Fachberichte 208, Berlin - Heidelberg - New York ... 1989, S. 278-287.

von Martial (1991a)
von Martial,F.; Victor,F.: Interaktive Planung von Bürovorgängen; in: Lutze,R.; Kohl,A. (Hrsg.): Wissensbasierte Systeme im Büro - Ergebnisse aus dem WISDOM-Verbundprojekt, München - Wien 1991, S. 313-324.

von Zimmermann (1990)
von Zimmermann,P.: Einsatz von objektorientierter Softwaretechnologie im Rechnungswesen; in: Scheer,A.-W. (Hrsg.): Rechnungswesen und EDV, 11. Saarbrücker Arbeitstagung 1990, Heidelberg 1990, S. 235-264.

Walther,C. (1987)
Walther,C.: A Many-Sorted Calculus Based on Resolution and Paramodulation, London - Los Altos 1987.

Wand (1989)
Wand,Y.: A Proposal for a Formal Model of Objects; in: Kim,W.; Lachovsky,F.H. (Hrsg.): Object-Oriented Concepts, Databases, and Applications, New York - Reading - Menlo Park ... 1989, S. 537-559.

Weber,A. (1987)
Weber,A.; Wiegand,S.: Integration von Produktionssystemen in höhere Petri-Netze, (Gemeinschafts-)Diplomarbeit, Fachbereich Informatik, Universität Dortmund, Dortmund 1987.

Weck (1982)
Weck,M.: Werkzeugmaschinen, Bd. 3: Automatisierung und Steuerungstechnik, 2. Aufl., Düsseldorf 1982.

Wedekind (1979a)
Wedekind,H.: Die Objekttypen-Methode beim Datenbankentwurf - dargestellt am Beispiel von Buchungs- und Abrechnungssystemen; in: Zeitschrift für Betriebswirtschaft, 49. Jg. (1979), S. 367-387.

Wedekind (1981)
Wedekind,H.: Datenbanksysteme I - Eine konstruktive Einführung in die Datenverarbeitung in Wirtschaft und Verwaltung, 2. Aufl., Mannheim - Wien - Zürich 1981.

Wedekind (1986a)
Wedekind,H.: Integrierte Fertigungsdatenbanken; in: Hommel,G.; Schindler,S. (Hrsg.): GI - 16. Jahrestagung I, Informatik-Anwendungen - Trends und Perspektiven, Proceedings, 6.-10.10.1986 in Berlin, Informatik-Fachberichte 126, Berlin - Heidelberg - New York - London - Paris - Tokyo 1986, S. 90-108.

Wedekind (1986b)
Wedekind,H.: Eine konzeptionelle Basis für den Einsatz von Datenbanken in Flexiblen Fertigungssystemen, Paper, Informatik VI, Universität Erlangen-Nürnberg, Erlangen o.J. (1986).

Wedekind (1987a)
Wedekind,H.; Zörntlein,G.: Eine konzeptionelle Basis für den Einsatz von Datenbanken in Flexiblen Fertigungssystemen; in: Informatik Forschung und Entwicklung, 2. Jg. (1987), S. 83-96.

Wedekind (o.J.a)
Wedekind,H.: Eine konzeptionelle Basis für den Einsatz von Datenbanken in Flexiblen Fertigungssystemen, Paper, Informatik VI, Universität Erlangen-Nürnberg, Erlangen o.J. (Inhaltlich nicht identisch mit Wedekind (1986b).)

Wendt (1989)
Wendt,S.: Nichtphysikalische Grundlagen der Informationstechnik - Interpretierte Formalismen, Berlin - Heidelberg - New York ... 1989.

Winter,Ro. (1991)
Winter,Ro.: Mehrstufige Produktionsplanung in Abstraktionshierarchien auf der Basis relationaler Informationsstrukturen, Dissertation, Universität Frankfurt 1989, Berlin - Heidelberg - New York ... 1991.

Zelewski (1986a)
Zelewski,S.: Das Leistungspotential der Künstlichen Intelligenz - Eine informationstechnisch-betriebswirtschaftliche Analyse, Band 1, 2 und 3, Dissertation (unter dem Titel: Das Leistungspotential der Künstlichen Intelligenz - Bestandsaufnahme und Bewertungsansätze aus informationstechnisch-betriebswirtschaftlicher Perspektive unter besonderer Berücksichtigung produktionswirtschaftlicher Aspekte -), Universität Köln 1985, Witterschlick (Bonn) 1986.

Zelewski (1986b)
Zelewski,S.: Netztheoretische Ansätze zur Konstruktion und Auswertung von logisch fundierten Problembeschreibungen, Arbeitsbericht Nr. 11 (2. Aufl. des Arbeitsberichts 8/1986), Seminar für Allgemeine Betriebswirtschaftslehre, Industriebetriebslehre und Produktionswirtschaft, Universität Köln, Köln 1986.

Zelewski (1989c)
Zelewski,S.: Petrinetze für die Konstruktion und Konsistenzanalyse von logisch orientierten Problembeschreibungen, Arbeitsbericht Nr. 28, Seminar für Allgemeine Betriebswirtschaftslehre, Industriebetriebslehre und Produktionswirtschaft, Universität Köln, Köln 1989.

Zelewski (1989e)
Zelewski,S.: Contributions of Net-Theory to the Modelling of OR-Problems from a Logically Based Point of View; in: Rivista di matematica per le scienze economiche e sociali, Anno 12 (1989), Fascicolo 2, S. 67-92.

Zelewski (1990c)
Zelewski,S.: Enterprise-Wide Data Modelling Information Systems in Industry (Buchbesprechung); in: Information Management, 5. Jg. (1990), Heft 2, S. 78-80.

Zhang,D. (1989)
Zhang,D.; Murata,T.: Fixpoint Semantics for Petri Net Model of Definite Clause Logic Programs, überarbeitete Fassung von: Technical Report UIC-EECS87-2, Department of Electrical Engineering and Computer Science, University of Illinois, Chicago 1987, Neufassung: Sacramento - Chicago o.J. (1989). (Anmk. des Verf.: erscheint in: Keener,L. (Hrsg.): Advances in the Theory of Computation and Computational Mathematics, Vol. 1, Norwood 1991.)

Institut für Produktionswirtschaft und Industrielle Informationswirtschaft

der Universität Leipzig

Verzeichnis der Arbeitsberichte

Nr. 1: ZELEWSKI, STEPHAN: Das Konzept technologischer Theorietransformationen - eine Analyse aus produktionswirtschaftlicher Perspektive, Leipzig 1994.

Nr. 2: SIEDENTOPF, JUKKA: Anwendung und Beurteilung heuristischer Verbesserungsverfahren für die Maschinenbelegungsplanung - Ein exemplarischer Vergleich zwischen Neuronalen Netzen, Simulated Annealing und genetischen Algorithmen, Leipzig 1994.

Nr. 3: ZELEWSKI, STEPHAN: Unternehmenskrisen und Konzepte zu ihrer Bewältigung, Leipzig 1994.

Nr. 4: SIEDENTOPF, JUKKA: Ein effizienter Scheduling-Algorithmus auf Basis des Threshold Accepting, Leipzig 1995.

Nr. 5: ZELEWSKI, STEPHAN: Petrinetzbasierte Modellierung komplexer Produktionssysteme (Projekt PEMOPS), Band 1: Exposition, Leipzig 1995.

Nr. 6: ZELEWSKI, STEPHAN: Petrinetzbasierte Modellierung komplexer Produktionssysteme (Projekt PEMOPS), Band 2: Bezugsrahmen, Leipzig 1995.

Nr. 7: ZELEWSKI, STEPHAN: Petrinetzbasierte Modellierung komplexer Produktionssysteme (Projekt PEMOPS), Band 3: Einführung in Stelle/Transition-Netze, Leipzig 1995.

Nr. 8: ZELEWSKI, STEPHAN: Petrinetzbasierte Modellierung komplexer Produktionssysteme (Projekt PEMOPS), Band 4: Verfeinerungen von Stelle/Transition-Netzen, Leipzig 1995.

Nr. 9: ZELEWSKI, STEPHAN: Petrinetzbasierte Modellierung komplexer Produktionssysteme (Projekt PEMOPS), Band 5: Einführung in Synthetische Netze, Teilband 5.1: Darstellung des Kernkonzepts, Leipzig 1995.

Nr. 10: ZELEWSKI, STEPHAN: Petrinetzbasierte Modellierung komplexer Produktionssysteme (Projekt PEMOPS), Band 5: Einführung in Synthetische Netze, Teilband 5.2: Auswertungsmöglichkeiten, Leipzig 1995.

Nr. 11: ZELEWSKI, STEPHAN: Petrinetzbasierte Modellierung komplexer Produktionssysteme (Projekt PEMOPS), Band 6: Erweiterungen von Synthetischen Netzen, Leipzig 1995.

Nr. 12: ZELEWSKI, STEPHAN: Petrinetzbasierte Modellierung komplexer Produktionssysteme (Projekt PEMOPS), Band 7: Fallstudie, Leipzig 1995.

Nr. 13: ZELEWSKI, STEPHAN: Petrinetzbasierte Modellierung komplexer Produktionssysteme (Projekt PEMOPS), Band 8: Charakterisierung des Petrinetz-Konzepts, Leipzig 1995.

Nr. 14: ZELEWSKI, STEPHAN: Petrinetzbasierte Modellierung komplexer Produktionssysteme (Projekt PEMOPS), Band 9: Beurteilung des Petrinetz-Konzepts, Leipzig 1995.

Nr. 15: ZELEWSKI, STEPHAN: Petrinetzbasierte Modellierung komplexer Produktionssysteme (Projekt PEMOPS), Band 10: Petrinetz-Literatur, Leipzig 1995.

Nr. 16: SIEDENTOPF, JUKKA: An Efficient Scheduling Algorithm Based upon Threshold Accepting, Leipzig 1995.

Nr. 17: SIEDENTOPF, JUKKA: The Threshold Waving Algorithm for Job Shop Scheduling, Leipzig 1995.

Nr. 18: ZELEWSKI, STEPHAN: Diskussionspapier zum Text "Zur wirtschaftlichen und sozialen Lage in Deutschland" einer evangelisch-katholischen Arbeitsgruppe, Leipzig 1995.

Nr. 19: SCHIMMEL, KATRIN; ZELEWSKI, STEPHAN: Untersuchung alternativer Auktionsformen hinsichtlich ihrer Eignung zur Koordination verteilter Agenten auf Elektronischen Märkten, Leipzig 1996.

Nr. 20: SIEDENTOPF, JUKKA: Feinterminierung unter restriktiven Laufzeitanforderungen - Ein exemplarischer Vergleich lokaler Suchverfahren (Teil I), Leipzig 1996.

Nr. 21: ZELEWSKI, STEPHAN: Strukturalistische Rekonstruktion von ökologisch induzierten Entwicklungen der produktionswirtschaftlichen Theoriebildung, Leipzig 1996.

Nr. 22: RÖßLER, HENRIK; SCHIMMEL, KATRIN: Zur Animation und Simulation hierarchischer Petrinetze., Leipzig 1996.

Nr. 23: RÖßLER, HENRIK; WURCH, MAIK: Implementierung des Modells eines Flexiblen Fertigungssystems, Teilbände 1-3, Leipzig 1996.

Nr. 24: SCHIMMEL, KATRIN: Abstimmung der Implementierungssoftware INCOME/STAR. Bericht zu Phase 1 des Projekts PEMVEK, Leipzig 1996/ 2. Auflage 1997.

Nr. 25: WURCH, MAIK: Modellierung eines Flexiblen Fertigungssystems sowie von Produktionsaufträgen. Bericht zu den Phasen 2 und 3 des Projekts PEMVEK, Leipzig 1996.

Nr. 26: SCHIMMEL, KATRIN: Der Einsatz elektronischer Märkte zur Koordination in Flexiblen Fertigungssystemen, Leipzig 1996.

Nr. 27: TÖPFER, ANDREAS: Vergleichende Wirtschaftlichkeitsbetrachtung von Windkraftanlagen im Raum Halle/Leipzig - Ergebniszusammenfassung, Leipzig 1996.

Nr. 28: WURCH, MAIK: Implementierung von Vickrey-Auktionen mit Hilfe von Petrinetzen, Leipzig 1996.

Nr. 29: WURCH, MAIK: Coordinating Electronic Markets by Auctions, Leipzig 1996.

Nr. 30: SCHIMMEL, KATRIN; WURCH, MAIK: Simulation eines Koordinations-Moduls in einem Flexiblen Fertigungssystem, Leipzig 1996.

Nr. 31: RÖßLER, HENRIK: XPNC - Auswahltool für parallele Schaltentscheidungen bei der Simulation von Petrinetzen, Leipzig 1997.

Nr. 32: ZELEWSKI, STEPHAN: Handelsinformationssysteme - erweiterte Fassung einer Rezension, Leipzig 1997.

Nr. 33: ZELEWSKI, STEPHAN: Erfahrungen mit Höheren Petrinetzen bei der Modellierung von Prozeßkoordinierungen in komplexen Produktionssystemen. Bericht zu Phase 7 des Projekts PEMVEK, Leipzig 1997. [in Arbeit]

Nr. 34: ZELEWSKI, STEPHAN: Optimierung in Petrinetz-Modellen - eine Analyse aus betriebswirtschaftlicher Sicht, Leipzig 1997. [in Arbeit]

Nr. 35: WURCH, MAIK: Simulation von Koordinations-Modulen unter Berücksichtigung strategischen Agentenverhaltens, Leipzig 1997. [in Arbeit]

Nr. 36: SCHIMMEL, KATRIN: Komponente für Erreichbarkeitsanalysen. Bericht zu Phase 6 des Projekts PEMVEK, Leipzig 1997.

Nr. 37: WURCH, MAIK: Modellierung der Prozeßkoordinierung. Bericht zu Phase 4 des Projekts PEMVEK, Leipzig 1997. [in Arbeit]

Nr. 38: BODE, JÜRGEN; FUNG, RICHARD Y.K.: Integrating Cost Considerations in Quality Function Deployment, Leipzig 1997.