

**Zeitschrift:** Bulletin des Schweizerischen Elektrotechnischen Vereins, des Verbandes Schweizerischer Elektrizitätsunternehmen = Bulletin de l'Association suisse des électriciens, de l'Association des entreprises électriques suisses

**Herausgeber:** Schweizerischer Elektrotechnischer Verein ; Verband Schweizerischer Elektrizitätsunternehmen

**Band:** 86 (1995)

**Heft:** 9

**Artikel:** Modellieren mit Specs-Netzen

**Autor:** Mattmann, Rudolf

**DOI:** <https://doi.org/10.5169/seals-902438>

### **Nutzungsbedingungen**

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften auf E-Periodica. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. Das Veröffentlichen von Bildern in Print- und Online-Publikationen sowie auf Social Media-Kanälen oder Webseiten ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. [Mehr erfahren](#)

### **Conditions d'utilisation**

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. La reproduction d'images dans des publications imprimées ou en ligne ainsi que sur des canaux de médias sociaux ou des sites web n'est autorisée qu'avec l'accord préalable des détenteurs des droits. [En savoir plus](#)

### **Terms of use**

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. Publishing images in print and online publications, as well as on social media channels or websites, is only permitted with the prior consent of the rights holders. [Find out more](#)

**Download PDF:** 28.01.2026

**ETH-Bibliothek Zürich, E-Periodica, <https://www.e-periodica.ch>**

Specs-Netze, eine Erweiterung der klassischen Petri-Netze, haben sich in der Praxis als Hilfsmittel für die Modellierung von Systemen unterschiedlichster Art bewährt. Der vorliegende Beitrag erläutert anhand einer Raumheizung die Grundlagen von Specs-Netzen. Die Modellierung eines Übertragungsprotokolls zeigt die Anwendung auf ein verteiltes System. Als weiteres Beispiel aus der Praxis wird die Simulation einer modernen Telefonstation besprochen.

# Modellieren mit Specs-Netzen

■ Rudolf Mattmann

Specs-Netze (von engl. specifications) sind ein grafisches Ausdrucksmittel, mit denen sich Echtzeitsysteme<sup>1</sup>, die parallele Prozesse beinhalten, veranschaulichen lassen. Mit Specs-Netzen kann die zeitliche Abhängigkeit der Prozesse untereinander und der Datenaustausch zwischen den Prozessen grafisch dargestellt werden. Zum Beispiel lassen sich damit potentielle Deadlocks automatisch erkennen. In der Praxis verwendet man Simulatoren, mit denen sich Specs-Netze grafisch auf dem Bildschirm simulieren und animieren lassen. Neu an den hier vorgestellten Specs-Netzen (siehe Kasten) sind ihre speziellen Eingabe-Ausgabe-Elemente, über die ein Netz während der Simulation mit seiner technischen Umgebung kommunizieren kann. Mit der grafischen Visualisierung dieser E-A-Elemente lassen sich Geräteoberflächen auf dem PC simulieren. Mittels eines Codegenerators lassen sich die Netze nach erfolgreicher Simulation in lauffähigen Ansi-C-Code übersetzen, der auf einem Mikrocontroller mit oder ohne Multitasking-Echtzeitbetriebssystem ausgeführt werden kann.

Die Entwicklung von eingebetteten Systemen mit der Specs-Netz-Methode steht in Konkurrenz zu strukturierten Case-Methoden. Wie bei diesen Methoden modelliert man auch bei Specs-Netzen in der Analysephase den Datenfluss, den Kontrollfluss und die Daten. Zusätzlich kann man jedoch auch noch die Bedienoberfläche des zu entwickelnden Gerätes oder der Anlage modellieren. Nach der Analysephase verlässt man bei den Case-Werkzeugen die

Datenflussmethode und wechselt zu einer Methode, mit der die Struktur der Softwaremodule und Prozeduren definiert werden. Dies bedeutet, dass man in der Entwurfsphase wieder zum Anfang zurückkehrt. Nicht so bei der Specs-Netz-Methode; bei dieser ergänzt man die in der Analysephase erstellten Netze durch Unternetze. Man verfeinert also den Detaillierungsgrad der Netze mit einer weiteren Schicht von Specs-Netzen. Am Ende dieser Phase liegt ein simulierbares Specs-Netz-Modell vor, mittels dessen man das tatsächliche mit dem gewünschten Verhalten vergleichen kann. Dies ist mit keiner der strukturierten Case-Methoden möglich.

## Erste Anwendung: Raumheizung

Als Beispiel für ein eingebettetes System [1] wird im folgenden eine einfache Raumheizung mit Specs-Netzen modelliert. Das Modell besteht aus den Teilen für die Anlage (Ölheizung, Gebäude, Witterungseinflüsse), dem Rechner inklusive Software und Peripheriebausteinen sowie einer Bedienungseinheit, die aus Tasten, Knöpfen, Schieberegler und Anzeigen besteht. Diese drei Teilsysteme werden bei Specs-Netzen (Netz-)Module genannt und werden als schraffierte Quadrate (Bild 1a) dargestellt. Netz-Module stellen die Abstraktion eines unterliegenden Netzes auf einem höheren Abstraktionsniveau dar.

Das Bild stellt die oberste Netzebene eines einfachen Gebäudeheizsystems dar. Es zeigt, wie die einzelnen Systemteile miteinander gekoppelt sind. Die Pfeile stellen die Datenflussrichtung dar. Das Modul Regler beispielsweise schickt Befehle (Brenner Ein/Aus, Mischventilstellung) an die Anlage und nimmt von ihr Messwerte entgegen (Temperaturwerte, Rauchgasan-

### Adresse des Autors:

Rudolf Mattmann, Dipl. El.-Ing. ETH, Ivyteam, Alpenstrasse 9, 6304 Zug.

<sup>1</sup> Echtzeitsysteme (und eingebettete Systeme) sind Rechnersysteme, die Geräte oder Anlagen softwaremässig steuern.



# Was sind Specs-Netze?

Specs-Netze [2; 3] stellen eine Erweiterung der klassischen Petri-Netze [1; 2] dar. Wie diese eignen sie sich zur Beschreibung von Systemen mit parallelen Aktivitäten. Die statische Struktur des Systems wird durch einen gerichteten Graphen dargestellt. Durch diesen Graphen laufen Marken, welche dem aktuellen Zustand des Systems entsprechen. Bei Specs-Netzen tragen diese Marken einen Wert mit sich, der bei der Simulation erzeugt und verändert wird. Als weitere Ergänzung besitzen Specs-Netze sogenannte Eingabe- und Ausgabe-Transitionen. Mit ihnen können während einer Simulation Daten mit der Umgebung ausgetauscht werden. Damit lassen sich mittels Specs-Netzen eingebettete Computersysteme modellieren, die auf äussere Ereignisse (z. B. Unterschreiten einer Soll-Temperatur) mit einer Reaktion (Einschalten der Heizung) antworten müssen. Zeitliche Ereignisse werden in Specs-Netzen mit speziellen Timer-Transitionen modelliert.

## Specs-(Petri-)Netze

Specs-Netze stellen eine Erweiterung der klassischen Petri-Netze dar und bestehen wie diese aus vier Grundelementen:

- *Stellen* (oft auch Plätze genannt) dienen zur Ablage von Daten, die mit Marken modelliert werden. Stellen können nur Marken eines bestimmten Datentyps aufnehmen.
- *Transitionen* stellen Aktionen dar. Mit ihnen werden Datentransformationen oder Zustandsänderungen ausgedrückt. Transitionen können bei Bedarf mit einer Schaltbedingung und einer Schaltaktion beschriftet werden. Die Beschriftung erfolgt mit Specs-Lingua, einer Pascal-ähnlichen Programmiersprache. Transitionen werden dadurch schaltbar, dass auf all ihren Eingangsstellen eine Marke liegt.
- *Pfeile* verbinden Stellen und Transitionen zu einem Netz und stellen die Richtung von Daten- und Kontrollflüssen dar. Zum Referenzieren der Daten werden Pfeile mit Namen beschriftet.
- *Marken* zeigen den Ort und den Wert von strukturierten Daten an. Sie können dynamisch erzeugt und vernichtet werden und durch das Netz fließen. Marken zeigen zudem den Ort der Kontrolle an.

Im Vergleich zu den höheren Petri-Netzen weisen Specs-Netze als Strukturierungsmittel zusätzliche hierarchische Transitionen (Agencies) und hierarchische Stellen (Channels) auf. Damit lassen sich auch grosse und komplexe Netze übersichtlich gestalten. Zudem verwenden sie spezielle Ein-/Ausgabe-Transitionen, um die Kommunikation des Systems mit seiner physikalischen Umgebung zu modellieren.

## Eingabe- und Ausgabe-Transitionen

Eingabe-Transitionen werden erst beim Auftreten eines äusseren Ereignisses schaltbar; sie erzeugen dabei eine Marke, die das Ereignis näher charakterisiert. Ausgabe-Transitionen wandeln beim Schalten Marken in externe Ereignisse oder Signale um. Eine Eingabe-Transition (Bild 3) kann zum Beispiel mit einem Temperaturfühler, einer Tastatur oder einer seriellen Schnittstelle assoziiert sein. Das Ereignis bedeutet in diesen Fällen, dass ein neuer Temperaturwert erfasst worden ist, dass jemand eine Taste niedergedrückt hat oder dass ein Byte empfangen worden ist. Der Wert des Markenattributs entspricht nach dem Schalten der Höhe der Temperatur, dem Tastencode oder dem Ascii-Wert des Zeichens.

Um externe Signale auf dem Bildschirm zu simulieren, können den Eingabe- und Ausgabe-Transitionen grafische Visualisierungen zugeordnet werden. Mit ihnen lassen sich sogar Bedienoberflächen von komplexen Geräten nachbilden.

Eine Ausgabe-Transition (Bild 4) wird aktiviert, wenn ihre Eingangsstelle mit einer Marke belegt wird. Beim Schalten entfernt die Ausgabe-Transition die Marke von ihrer Eingangsstelle und erzeugt ein externes Signal, das dem Wert der entfernten Marke entspricht. Eine Ausgabe-Transition kann zum Beispiel mit einer Anzeige, einem Ventil oder einem Motor verbunden sein. Beim Schalten wird der Attributwert umgewandelt und als Zeichen auf der Anzeige dargestellt, die Ventilöffnung entsprechend angepasst, oder es wird die Spannung des Motors neu eingestellt.

teile). Das Modul Bediengerät ist mit den Regler- und Anlagemodulen verbunden. Von beiden Teilsystemen empfängt es Informationen, die auf einer Anzeige dargestellt werden. Die Frontansicht eines möglichen Bediengerätes zeigt Bild 2. Ein Benutzer kann am Schieber die Soll-Temperatur einstellen und rechts davon die momentane Raumtemperatur ablesen. Am Drehknopf lässt sich ein Parameter der Heizkurve einstellen.

Das Bild 1a zeigt neben der statischen Struktur des Gesamtsystems auch den momentanen Zustand während einer Simulation. Innerhalb der Kreise (genannt Stellen), die die Schnittstellen zwischen den Modulen bilden, erkennt man schwarze Punkte (genannt Marken), welche die Informationen, die über die Datenflusspfeile fließen, darstellen. Aus Bild 1a ersieht man, dass die Soll-Temperatur auf 20°C eingestellt ist, die Aussentemperatur 3°C und die Raumtemperatur 17°C beträgt. Während einer Simulation sieht man auf dem Bildschirm, wie Marken im Regelkreis zwischen Regler und Anlage kreisen und welches ihre Werte sind. Während der Simulation kann der Benutzer mit der Maus interaktiv die Knöpfe und Schieber auf dem Bediengerät betätigen und damit den Regler beeinflussen. Auf der Anzeige können stets die aktuellen Messwerte abgelesen werden.

Will man das Gesamtsystem simulieren, so muss man ein einfaches Netzmodell der Anlage erstellen. Das Bild 1b zeigt das Modul (Unternetz) Anlage. Auch dieses besteht wiederum aus Modulen, wobei eines das Heizsystem (z.B. eine Ölfuehrung), eines das Gebäude und eines die Umgebung repräsentiert. Während das Modul Gebäude die Informationen über die Wärmedämmung des Hauses kapselt, berücksichtigt das Modul Umgebung die Witterungseinflüsse. Untereinander sind die Module über Stellen gekoppelt. Man sieht zudem die Schnittstelle der Module zur oberen Netzebene; sie werden durch gestrichelte Kreise symbolisiert, die die gleichen Anschriften tragen wie ihre (ungerasteten) Repräsentanten auf der oberen Ebene.

Betrachten wir als Beispiel ein Netzteil, wie es im Modul Regler vorkommen könnte (Bild 5a). Man sagt, die Transition ist schaltbar, wenn auf jeder Stelle, die mit einem Datenflusspfeil in Richtung Transition verbunden ist, eine Marke liegt und wenn gleichzeitig die der Transition zugeordnete Schaltbedingung erfüllt ist. Schaltbare Transitionen (erkennbar am schwarzen Quadrat innerhalb der Transition) können schalten. Dabei verschieben sie die Marken von ihren Eingangsstellen auf ihre Ausgangsstellen (Bild 5b). Gleichzeitig führen die Transitionen den ihnen zugeordneten Programmcode aus und berechnen



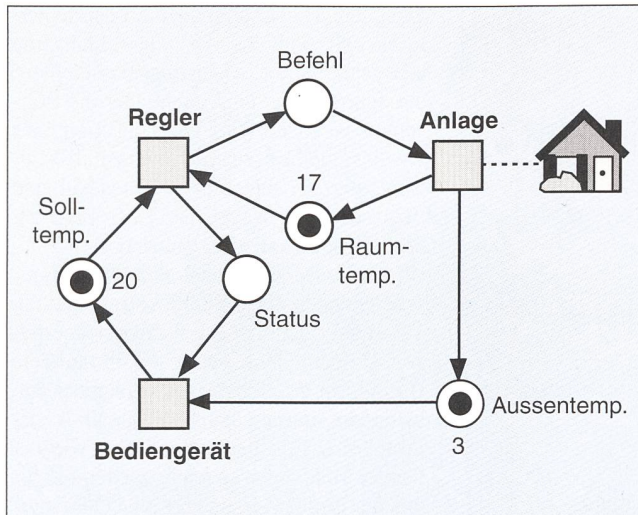


Bild 1a Specs-Netz Raumheizung

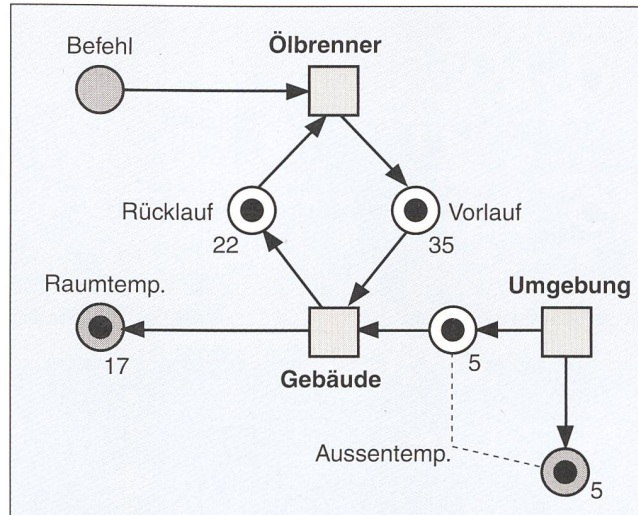


Bild 1b Modul Anlage

damit die Werte der Ausgangsmarken. Das Verschieben der Marken und das Ausführen des Programmcodes bilden eine nicht unterbrechbare Aktion.

In einem Specs-Netz ist es durchaus üblich, dass mehrere Transitionen gleichzeitig schaltbar sind. Bild 6, in dem mehrere Netzebenen sichtbar gemacht wurden, zeigt eine Situation, in der zwei Transitionen schaltbar sind. Da die beiden Transitionen miteinander nur lose gekoppelt sind, also keine Stellen miteinander teilen, dürfen sie gleichzeitig oder, wie man auch sagt, parallel oder nebenläufig schalten. Würden die Transitionen hingegen mindestens eine Stelle miteinander teilen, wäre dies nicht möglich, da Transitionen in einem unteilbaren Schritt schalten. Sie würden in einer solchen Konstellation nicht gleichzeitig, sondern nacheinander schalten.

Das Dreieck in Bild 7 stellt Specs-Netze in Beziehung zu drei verwandten Modellen der Informatik: Datenflussgraphen, Zu-

standsmaschinen und mathematischen Funktionen. Die Grafiken der Specs-Netze, einschliesslich der Strukturierung in eine Hierarchie von Teilnetzen, sind mit hierarchischen Datenflussgraphen verwandt, wie sie in vielen Case-Methoden [5;6] verwendet werden. Die Netzgrafik gliedert Systeme in Teilsysteme, stellt ihre Kopplung dar und beschreibt den Informationsfluss zwischen ihnen. Die ausführbare Semantik der Specs-Netze ist eng verwandt mit der Semantik von gekoppelten Zustandsmaschinen [7]. (Ein Beitrag zum Werkzeug CIP, das auf gekoppelten Zustandsmaschinen beruht, wird in Bulletin SEV/VSE 17/95 erscheinen.) Der Zustand eines Specs-Netzes ist durch die momentane Verteilung der Marken im Netz gegeben. Schliesslich erfolgt die Beschriftung des Specs-Netzes in einer Sprache, die eng an Pascal angelehnt ist und mit der sich Funktionen und Datenstrukturen ausdrücken lassen.

Specs-Netze unterscheiden sich von ähnlichen Formalismen wie State Charts [8] oder SA/RT [5; 6] dadurch, dass Specs-Netze den Kontroll- und Datenfluss nicht trennen, sondern gleichwertig behandeln. In Specs-Netzen werden Daten mit Marken symbolisiert, die Informationen tragen und in Form von Werten oder komplexen Datenstrukturen vorliegen können. In Specs-Netzen wird auch die Kontrolle mit Marken modelliert, wobei in erster Linie nicht deren Datenwert, sondern ihre An- und Abwesenheit bestimmend sind. Transitionen werden ja dadurch schaltbar, dass auf all ihren Eingangsstellen eine Marke liegt.

Dank der sorgfältigen Verknüpfung dreier mächtiger Informatikmodelle eröffnet sich den Specs-Netzen ein grosses Anwendungsfeld der technischen Informatik. Wie wir im ersten Anwendungsbeispiel gesehen haben, eignen sich Specs-Netze, um eingebettete Systeme in ihrer Gesamtheit zu modellieren. Dies ist für die Simulation sehr wichtig, da man die Software eines zu entwickelnden Systems nicht isoliert austesten kann. Dazu müssen sowohl die Benutzerschnittstelle als auch ein minimales Modell der Umgebung simuliert werden.

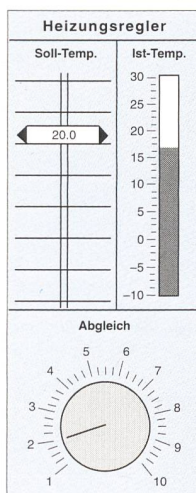


Bild 2 Bediengerät

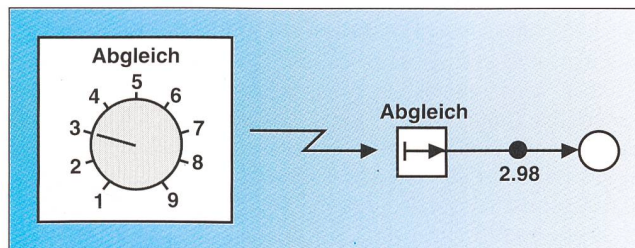


Bild 3 Eingabe-Transition

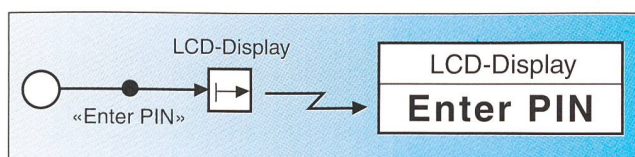


Bild 4 Ausgabe-Transition

## Weitere Anwendungen

### Protokollsimulation

Um dem Leser die Funktionsweise der Specs-Netze näherzubringen, wird im folgenden Abschnitt ein einfaches Protokollbeispiel besprochen. Bild 8a zeigt, wie zwei Stationen – ein Sender und ein Empfänger – über eine bidirektionale Kommunikationsleitung kommunizieren. In der einen Richtung arbeite die Datenübertragung nur zu 99% zuverlässig, während der andere Kanal zuverlässig sei. Wir treffen dabei die Annahme, dass der Empfänger in der Lage ist, mit Hilfe einer Prüfsumme die



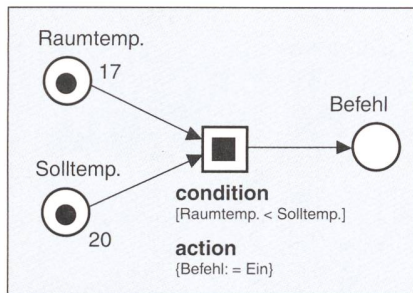


Bild 5a Schaltbare Transition

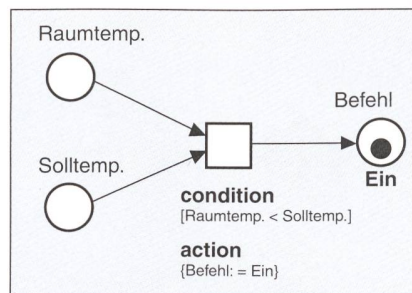


Bild 5b Transition nach dem Schalten

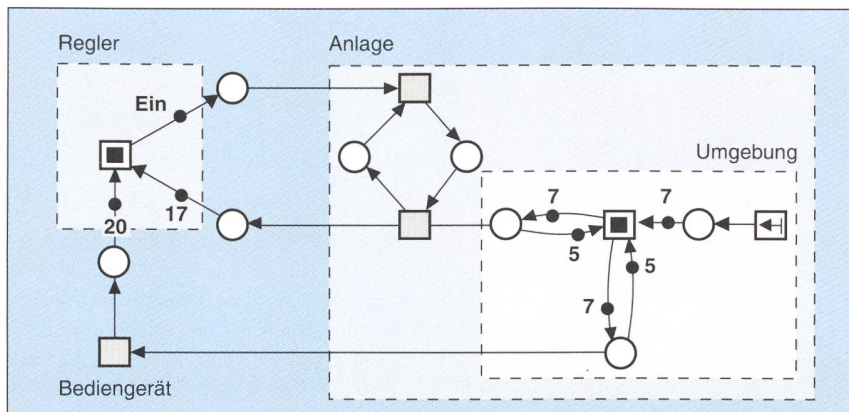


Bild 6 Paralleles Schalten

### Zur Entwicklungsgeschichte von System-Specs

Die Grundlagen zu einem grafischen Petri-Netz-Werkzeug sind 1985 bis 1987 in einem KWF-Projekt (Kommission zur Förderung der wissenschaftlichen Forschung; Projekt-Nr. 1498) gelegt worden, das vom Bund und der Industrie (Zellweger, Hasler, Landis & Gyr) getragen wurde. Daraus entwickelte sich das in [4] beschriebene Werkzeug Pace, ein Petri-Netz-Werkzeug, dessen Netze mit Smalltalk beschriftet werden. Zwischen 1988 und 1993 wurden parallel dazu die mit Pascal beschrifteten Specs-Netze und das Werkzeug System-Specs im Forschungslabor von Landis & Gyr entwickelt. Die Entwicklungsgruppe gründete 1993 die Aktiengesellschaft Ivyteam und übernahm von Landis & Gyr alle Rechte am Produkt. Seither werden System-Specs und das Schwesterprodukt Business-Specs von Ivyteam zusammen mit Partnerfirmen weiterentwickelt und europaweit vermarktet. Die laufenden Forschungsarbeiten werden vom schweizerischen Schwerpunktprogramm Informatikforschung (Nr. 5003-34299) mitgetragen und stellen eine Zusammenarbeit mit der Universität Bern und der ETH Lausanne dar.

korrekte oder fehlerhafte Übertragung einer Meldung zu beurteilen. Das Protokoll sieht vor, dass der Sender die Meldungen einzeln überträgt und jeweils auf ein Quittierungssignal wartet. Im Falle einer positiven Quittung schickt der Sender sogleich die nächste Meldung. Bei einer negativen Quittung wiederholt er die zuletzt geschickte Meldung. Dieses sehr einfache Senderprotokoll kann, wie Bild 8b zeigt, mit zwei Transitionen und einer internen Stelle gelöst werden. Die Transition auf der linken Seite nimmt von der oberen Netzebene eine Meldungsmarke entgegen, legt davon eine Kopie auf der internen Stelle Buffer ab und eine zweite, ergänzt mit einer Prüfsumme, auf der Schnittstellen-Stelle, die mit dem unzuverlässigen Kanal verbunden ist. Die Transition wird immer dann schaltbar, wenn von der Empfängerstation eine positive Quittungsmeldung (true) vorliegt. Im Fall, dass die Quittungsmeldung negativ (false) ist, schaltet die Transition auf der rechten Seite, indem sie eine Kopie der Marke von der internen Stelle Buffer liest und die Kopie zusammen mit der Prüfsumme nochmals dem Kanal übergibt. Bild 8c zeigt das Unternetz des Empfängerprotokolls, das eine ähnliche Struktur wie das Sendernetz aufweist, aber ohne interne Stelle auskommt. Eine Besonderheit weist das in Bild 8d dargestellte Netz des unzuverlässigen Kanals auf. Die beiden fett hervorgehobenen Transitionen sind für das Übertragen der Meldungen verantwortlich.

Die obere Transition modelliert den fehlerlosen Fall, während die untere Transition eine fehlerhafte Übertragung so modelliert, dass nur die Meldung, nicht aber die Prüfsumme verändert wird. Die Transition links unten modelliert einen Pseudozufallszahl-Generator, der eine Zahl zwischen Null und Eins generiert. Wegen der Bedingung auf den Eingangskanten der beiden oberen Transitionen wird der Kanal mit 99% Wahrscheinlichkeit die Meldung korrekt und mit 1% Wahrscheinlichkeit verändert übertragen. Die Simulation des Protokolls auf dem Bildschirm ermöglicht selbst Nichtspezialisten ein unmittelbares Verständnis des Protokolls. Der Betrachter sieht, wie der Sender Meldungen so lange wiederholt, bis ihm der Empfänger eine positive Quittungsmeldung schickt. Wie das Beispiel zeigt, ist es mit Specs-Netzen sehr einfach möglich, verteilte Applikationen und Kommunikationsprotokolle zu modellieren, zu simulieren und zu verifizieren. Unter Verifizieren versteht man die mathematische Überprüfung der geforderten Netzeigenschaften. Dank der formalen Definition der Specs-Netze war es möglich, Algorithmen zu entwickeln, die beweisen, dass das Netz nicht verklemmt und dass der Kanal zu jedem Zeitpunkt nur in einer Richtung benützt wird.

### Telefonstation

Als Beispiel für eine industrielle Anwendung zeigt der folgende Abschnitt, wie mit dem Werkzeug System-Specs [9] die Bedienungsführung und die Funktionalität einer neuen Generation von Kartentelefonstationen modelliert und simuliert worden ist. Die mit dem grafischen Editor erstellte Bedienoberfläche der Telefonstation und die Möglichkeit zur Auswahl und Parametrisierung einer Karte sind in den Bildern 9a, 9b und 9c dargestellt. Ein Mausklick auf ein Kartensymbol erzeugt in der zugeordneten Eingabetransition eine Marke, deren Wert dem mit dem Schieberegler eingestellten Geldbetrag entspricht. Die Marke wird vom Netz-Modul Kartenleser als Ereignis «Benutzer hat eine Karte eingeschoben» interpretiert. Die Gliede-

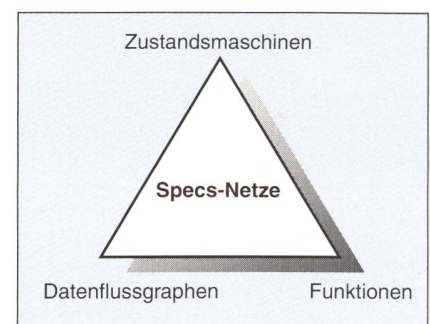


Bild 7 Einordnung der Specs-Netze in die klassischen Modelle der Informatik



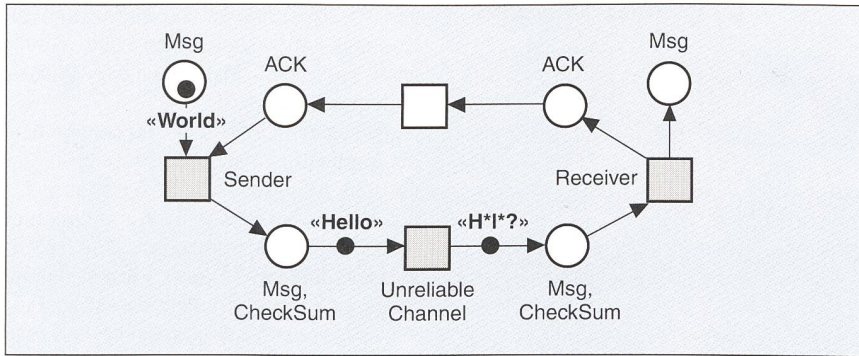


Bild 8a Specs-Netz Protokoll

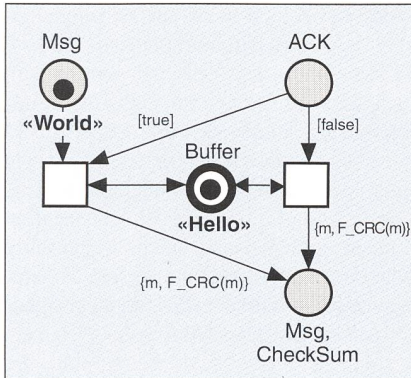


Bild 8b Modul Sender

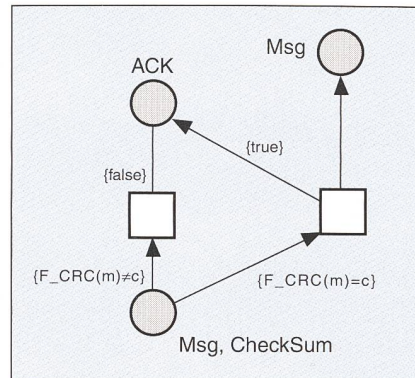


Bild 8c Modul Empfänger

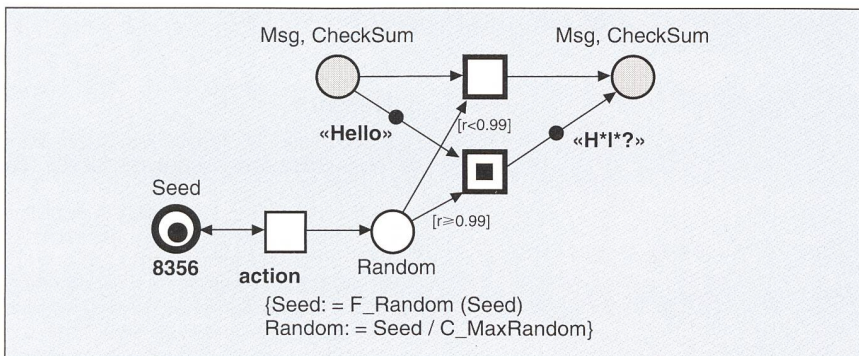


Bild 8d Modul Gestörter Kanal

zung des Specs-Netzes Telefonstation in Untermodule zeigt, dass das Lesen und Interpretieren von Karten, die Verarbeitung von Tasten, das Senden und Empfangen von Meldungen via Telefonleitung und die Ausgabe von Texten auf der Flüssigkristallanzeige von autonomen Geräte-modulen parallel ausgeführt werden. Im Zentrum des Netzes befinden sich drei Hauptmodule, die die Funktionalität der Telefonstation ausmachen. Je eines der Module ist verantwortlich für die Überprüfung der Kartenauthentizität, das Verarbeiten von Taximpulsen und die Programmierung von Chip-Karten. Der Zustand der Telefonstation als Gesamtes ist durch die Teilzustände der verschiedenen Unter-module gegeben. Der logische Betriebszustand der Telefonstation während eines Gesprächs wird jedoch explizit in einer speziellen Zustands-Marke verwaltet, die

zyklisch durch das Netz wandert und auf diese Weise die einzelnen Aktivitäten der Untermodule synchronisiert.

Ziel der Modellierungen mit Specs-Netzen war, das technische Pflichtenheft durch eine dynamische Spezifikation zu ergänzen. Es ging vor allem darum, für eine neue Generation von Kartentelefonstationen eine einfache Bedienungs-führung für die recht komplexe Funktionalität zu finden und zu definieren. Es zeigte sich, dass für die Modellierung gewisser Funktionen das Netz durch ein einfaches Modell einer Telefonzentrale und eines übergeordneten Verrechnungssystems ergänzt werden musste. Während das Telefonzentralenmodell die Aufgabe hatte, eine eintreffende Telefonnummer zu analysieren, die entsprechende Tarifstufe zu bestimmen und danach in fest vorgegebenen Zeitinter-vallen Taximpulse an die Telefonstelle zu

schicken, hatte das Verrechnungssystem zu überprüfen, dass die Karte nicht gesperrt und der korrekte Pin-Code eingegeben worden war. Als der so erstellte Prototyp dem Auftraggeber vorgeführt wurde, gelangte dieser zur Erkenntnis, dass dem Telefonbenutzer nicht zugemutet werden kann, untätig zu warten, bis die Telefonstation eine Verbindung zum Verrechnungssystem aufgebaut und dieses die Karte überprüft hat. Auftraggeber und Hersteller entschlossen sich, das Pflichtenheft der Bedienungs-führung abzuändern und die Menüführung so zu gestalten, dass die Kartenprüfungen im Hintergrund ablaufen und der Benutzer bereits mit Eintippen der Telefonnummer beginnen oder einen Menüpunkt der Spezial-funktionen auswählen kann.

Diese scheinbar geringfügige Änderung am Pflichtenheft hatte Einfluss auf viele andere Bedienungsabläufe. Dank der Verwendung des mit Specs-Netzen erstellten Prototyps konnte er von einer Person innert einer Woche an die neuen Anforderungen angepasst werden. Hätte die Bedienungs-führung jedoch erst am fertig entwickelten Gerät überprüft werden können, wären die Mängel vielleicht erst dann entdeckt worden. Dies hätte zu einem umfangreichen Redesign der gesamten Software geführt und das Umschreiben vieler Programm-module nach sich gezogen.

## Ausblick

Die Abstützung der Specs-Netze auf bewährte Modelle der Informatik erlaubt eine Verifikation des modellierten Systems

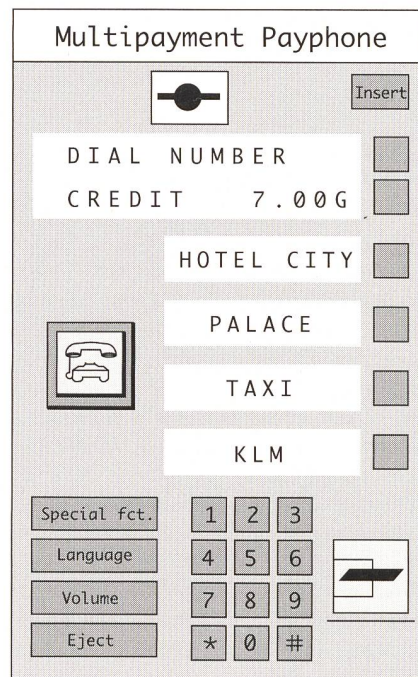


Bild 9a Telefonbedienung



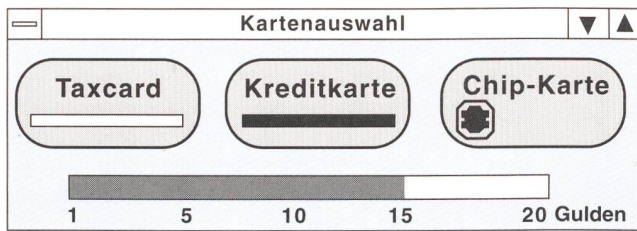


Bild 9b Kartenauswahl

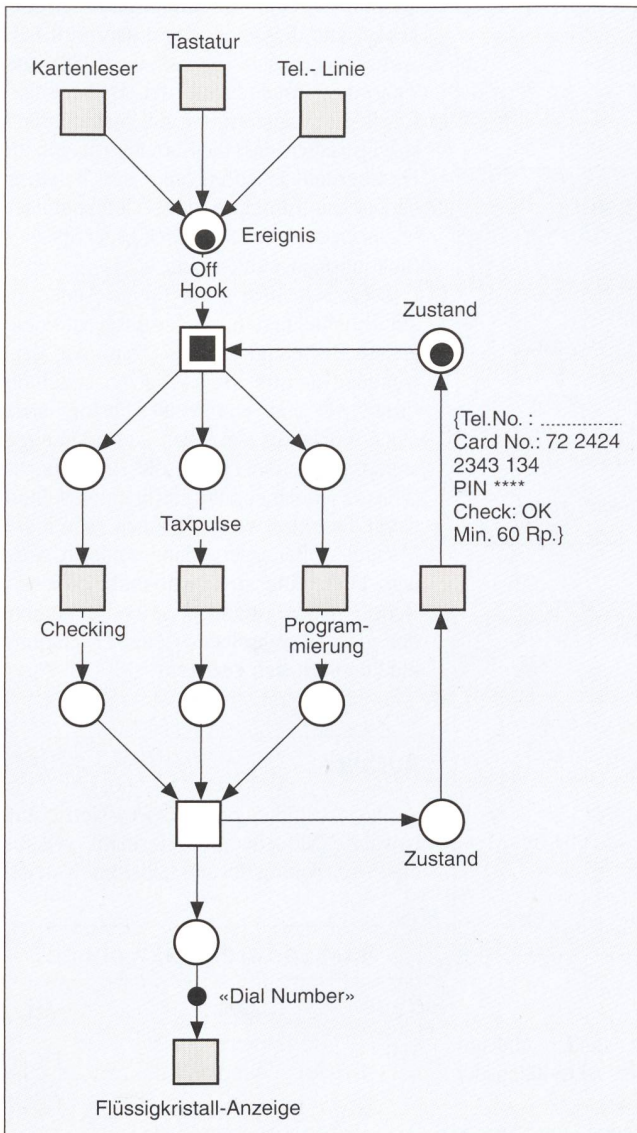


Bild 9c Specs-Netz Telefonstation

im mathematischen Sinne. Damit kann in vielen Fällen die technische Korrektheit nachgewiesen werden. Die Simulationmöglichkeiten von System-Specs, vor allem in Verbindung mit visualisierten Geräteoberflächen, ermöglicht eine Validierung des Produktes durch den Kunden, bevor es implementiert wird. Das garantiert, dass das aus der Sicht des Kunden «richtige» Produkt entwickelt wird. Dank der Verwendung der mathematisch fundierten Specs-Netze können neue Produkte anschaulich und exakt definiert, auf dem Bildschirm simuliert und mittels Codegeneratoren implementiert werden.

System-Specs wurde in so verschiedenen Anwendungen eingesetzt wie: Definition und Implementierung der Menüführung von Kartentelefonen, Analyse- und Regler-

geräten, Spezifikation und Simulation von Kommunikationsprotokollen und Simulation von Verkehrskreuzungen, Wirbelsäulen und Narkosen.

Wegen der Anschaulichkeit der animierten Markenflüsse auf dem Bildschirm eignet sich die Specs-Netz-Methode auch für den Einsatz bei der Analyse und Optimierung von Geschäftsprozessen. Um die Anwender optimal zu unterstützen, wurde Business-Specs entwickelt, das speziell erweiterte Specs-Netze verwendet und eine Bibliothek von häufig wiederverwendbaren Netzkomponenten enthält. Business-Specs eignet sich nicht nur zur Modellierung von Geschäftsvorfällen in Banken, Versicherungen und Verwaltungen, sondern auch zum Entwickeln neuer Restauranttypen, zum Planen der optimalen Auslastung von Bahnbetrieben oder zum Festlegen einer effizienten Anordnung von Warenstellen, damit sich in Einkaufszentren Kunden und Personal nicht gegenseitig behindern. Interessant in diesem Zusammenhang ist, dass die für Business-Specs nötigen Erweiterungen der Specs-Netze ein Resultat der Forschungsarbeiten sind, die letztes Jahr im Rahmen des schweizerischen Schwerpunktprogramms Informatik durchgeführt worden sind.

## Literatur

- [1] H.P. Gisiger, A. Kündig: Petri-Netze: Netze zur Modellierung verteilter Systeme. Bulletin SEV/VSE 79(1988)17, S. 1026-1034.
- [2] B. Büttler, R. Esser, R. Mattmann: A distributed simulator for high order Petri nets. Lecture Notes of Computer Science 483, Springer-Verlag 1991, S. 47-63.
- [3] Ivyteam: Systementwurf auf Anhieb richtig. Design & Elektronik 22/1994.
- [4] J. Dähler: Ein interaktives Simulations- und Programmengenerations-Werkzeug für erweiterte Petri-Netze. Bulletin SEV/VSE 79(1988)17, S. 1035 bis 1040.
- [5] E. Yourdon: Moderne strukturierte Analyse. Wolframs Fachverlag 1992.
- [6] P.T. Ward, S.J. Mellor: Structural Development for Real-Time Systems. Vol. 1-3, Yourdon Press, 1985.
- [7] H. Fierz, H. Müller, S. Netos: CIP-Communicating Interacting Processes. A Formal Method for the Development of Reactive Systems. Gorski J. (Ed.): Proceedings Safecom '93, Poland 1993, Springer-Verlag 1993.
- [8] D. Harel: On Visual Formalisms. Communications of the ACM 31(1988)5, pp. 514-530.
- [9] Ivyteam: System Specs 3.0, Reference Manual, 1995.

## Modéliser avec les réseaux Specs

Les réseaux Specs, une extension des réseaux de Pétri classiques, ont été utilisés pour modéliser des systèmes très divers. Cet article présente le formalisme des réseaux Specs à travers un réglage de chauffage. Ensuite l'article présente la modélisation d'un protocole de communication comme application distribuée. Enfin la simulation d'une nouvelle génération de téléphone à cartes est décrite.